昭和59年5月1日発行（毎月1回1日発行）第3巻5号通巻第24号　昭和58年6月7日国鉄首都特別扱承認雑誌第6897号　昭和58年11月2日第三種郵便物認可

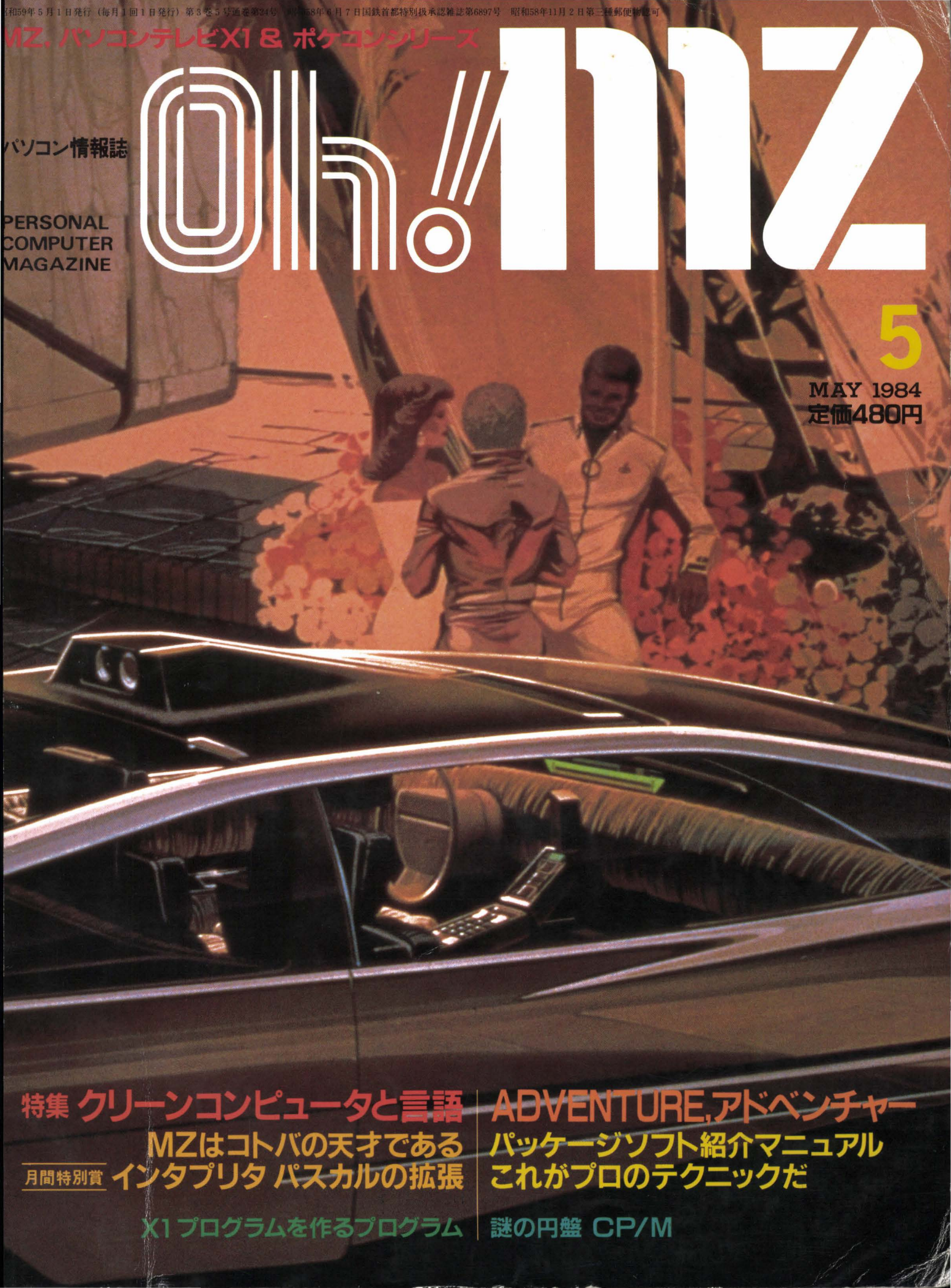
MZ, パソコンテレビX1 & ポケコンシリーズ

パソコン情報誌

Oh!MZ

PERSONAL
COMPUTER
MAGAZINE

5
MAY 1984
定価480円



特集 クリーンコンピュータと言語
MZはコトバの天才である
月間特別賞 インタプリタ パスカルの拡張
X1プログラムを作るプログラム

ADVENTURE,アドベンチャー
パッケージソフト紹介マニュアル
これがプロのテクニックだ
謎の円盤 CP/M

SHARP

PASCAL、LOGO、MACHINE…多彩なシス

倍精度BASIC

技術計算や事務処理計算に

10進演算方式による有効精度16桁。しかも単に数値データが倍精度になっているだけでなく、各種のアルゴリズム設計、データ処理などそれぞれに高機能を実現。同時に高速処理を指向した設計で、ビギナーはもちろん、高度な各種プロフェッショナルプログラミングも縦横に駆使でき、ビジネスにも存分に活かせます。

●MZ-1Z003(MZ-2200/2000用、テープバージョン) 標準価格 7,000円
●MZ-2Z003(MZ-2200/2000用、ディスクバージョン) 標準価格12,000円

インタープリタPASCAL

構造化プログラミング学習に

カセットテープベースで即実行型とするためインタープリタ方式を採用、BASICのように簡単に使えるPASCALです。再帰的記述や構造的アルゴリズム設計など標準PASCALの機能はほとんどもっており、構造化プログラミング学習に最適、これからPASCALを学ぼうという人にぜひ使っていただきたいソフトです。

●MZ-1Z004(MZ-2200/2000用、テープバージョン) 標準価格12,000円

マシンランゲージ

気軽にマシン語にアタック

マシン語のプログラムを入力したり、見たり、動かしたりできるマシン語モニタプログラム。MZ-2200/2000に付属のマシン語モニタの機能を強化した16のコマンドや簡単なデバッグ機能ももっています。コントロールキーの機能も強力で、マシン語サブルーチンの作り方などを学習中の人に最適な、使いやすいプログラムです。

●MZ-1Z006(MZ-2200/2000用、テープバージョン) 標準価格7,000円

シャープのMZは、各種言語やソフトウェアが自由に入れ換えできる"クリーン"設計。その特長を存分に発揮させるためのシステムソフトも各種サポート、自由自在のシステムチェンジでPASCALマシンにも開発マシンにも…。さらにMZ-2000で蓄積された膨大な数のアプリケーションソフトもそのまま使用できるなど、優れた市販ソフト環境を誇っています。MZ-2200なら多彩なコンピュータの世界が約束されています。

パーソナルコンピュータ

MZ-2200

標準価格128,000円

<MZ-2200の主な特長>●アドレス空間64Kバイト、オールRAM。応用自在のクリーンメモリシステム●本体だけでなく周辺機器をも含めたシステムコストパフォーマンスを徹底して追求した経済設計●高機能・高速CPU Z80A搭載●8色カラーコントロールをはじめとしたハイレベルなグラフィック機能●16ビットへの対応も考慮した先進設計●4スロットの拡張ユニット標準装備●操作性を重視した前面コントロール(IPLスイッチ、リセットスイッチ、音量ボリウム)

機種別、目的別ソフト・ハード満載!!
「MZアプリケーション」Vol.5 定価300円
●お求めは最寄りのMZ取扱店でどうぞ。

新作ソフト、周辺機器開発情報!!
「MZニューアプリケーションニュース」
●随時無料発行しています。最寄りのMZ取扱店でどうぞ。

資料請求券 MZ-2200 or 11 MZ 5係

シャープ株式会社 本社 〒545 大阪市阿倍野区長池町22番22号☎(06)621-1221(大代表)
●お問い合わせ、資料請求は…シャープ㈱国内産機営業本部 〒162 東京都新宿区市谷八幡町8番地☎(03)260-1161(大代表)

テムソフト、だからいつも新機種としてつき合えます。

システムプログラム

テープベースのマシン語開発ツール

マシン語プログラムを開発するためのアプリケーションパッケージで、27種のコマンドを持つエディタ/マクロアセンブラ・K/C、1200→2200/2000コンバータ、リンカ、22種のコマンドをもつシンボリックデバッガ、PROMフォーマッタといった、システム開発には欠かせない4本のテープから成っています。スクロール速度の指定、TABの設定・解除などコントロールキーの機能も強化されています。

●MZ-1Z005(MZ-2200/2000用、テープバージョン)標準価格25,000円

MZ-LOGO

新しい時代の言語

LOGO(ロゴ)は、アメリカで開発され、いま世界中で話題の新言語。60°曲がって5歩進め……とか、同じ四角をくり返せ……とか、複雑な図形も人間の考える通りにいとも簡単に描くことができ、初めてパソコンにふれる人にとって、もっともふさわしい言語だともいわれています。コンピュータの知識がなくても、気軽にプログラムが組めます。

●MZ-LOGO(MZ-2200/2000用、テープバージョン)9,800円
㈱日本ソフトバンクより発売中。

フロッピーDOS

本格的マシン語プログラムづくりに

ディスクによるプログラム開発を容易にする本格的ツール。Z80の命令をフルに利用し、メモリ効率やスピードもアップ。さらにスクリーンエディタ、Z80マクロアセンブラ、リンカ、デバッガ、PROMフォーマッタをもつ他、実数型と整数型のふたつのBASICコンパイラという高水準ソフトも付属。ファイル操作のコマンドも豊富、またプログラム開発には恰好のユーティリティも数多く備えています。

●MZ-2Z004(MZ-2200/2000用、ディスクバージョン)標準価格50,000円

漢字カラーディスクBASIC

より高度なシステム活用に

MZ-2200/2000のカラーグラフィック、漢字コントロール機能を十分に発揮させ、高速処理と強力なファイルコントロールを可能にした待望のBASICインタープリタです。KINPUT命令によるカナ漢字変換、またKPRINT/PRINTを使用することにより、同一行にキャラクタと漢字の混在印字も可能。さらにDELETE、RENUM SEARCHなど追加・拡張コマンドも豊富で、より使いやすくなっています。

●MZ-2Z021(MZ-2200/2000用、ディスクバージョン)標準価格5,000円
※漢字ROMボードMZ-1R13が必要です。

●写真の12型グリーンディスプレイMZ-1D12、14型カラーディスプレイMZ-1D15、データレコーダMZ-1T02、フロッピーディスクドライブMZ-1F07、ドットプリンタMZ-1P07はオプションです。●テープベースでMZ-2200をご使用の場合はデータレコーダMZ-1T02が必要です。

信頼のハードと強力なソフトで実務領域を拡げる実力機

パーソナルコンピュータ
MZ-3500 シリーズ

MZ-3531 標準価格320,000円
(ミニフロッピー1基内蔵)
MZ-3541 標準価格410,000円
(ミニフロッピー2基内蔵)

●写真は本体(MZ-3541)、キーボード(MZ-1K06標準価格38,000円)、CRT(MZ-1D03標準価格163,000円)を組合せた例です。※画面はオプションのグラフィックボード、グラフィックメモリ(×2)を使用した例です。
●画面はハメコミ合成写真です。

豊富なソフトのロングセラーMZ

パーソナルコンピュータ
MZ-700 シリーズ

MZ-711……標準価格 79,800円
MZ-721……標準価格 89,800円
(データレコーダ内蔵)
MZ-731……標準価格 128,000円
(データレコーダ・カラープロッタプリンタ内蔵)

●写真はMZ-731とCRT(MZ-1D05標準価格69,800円)を組合せた例です。
●画面はハメコミ合成写真です。

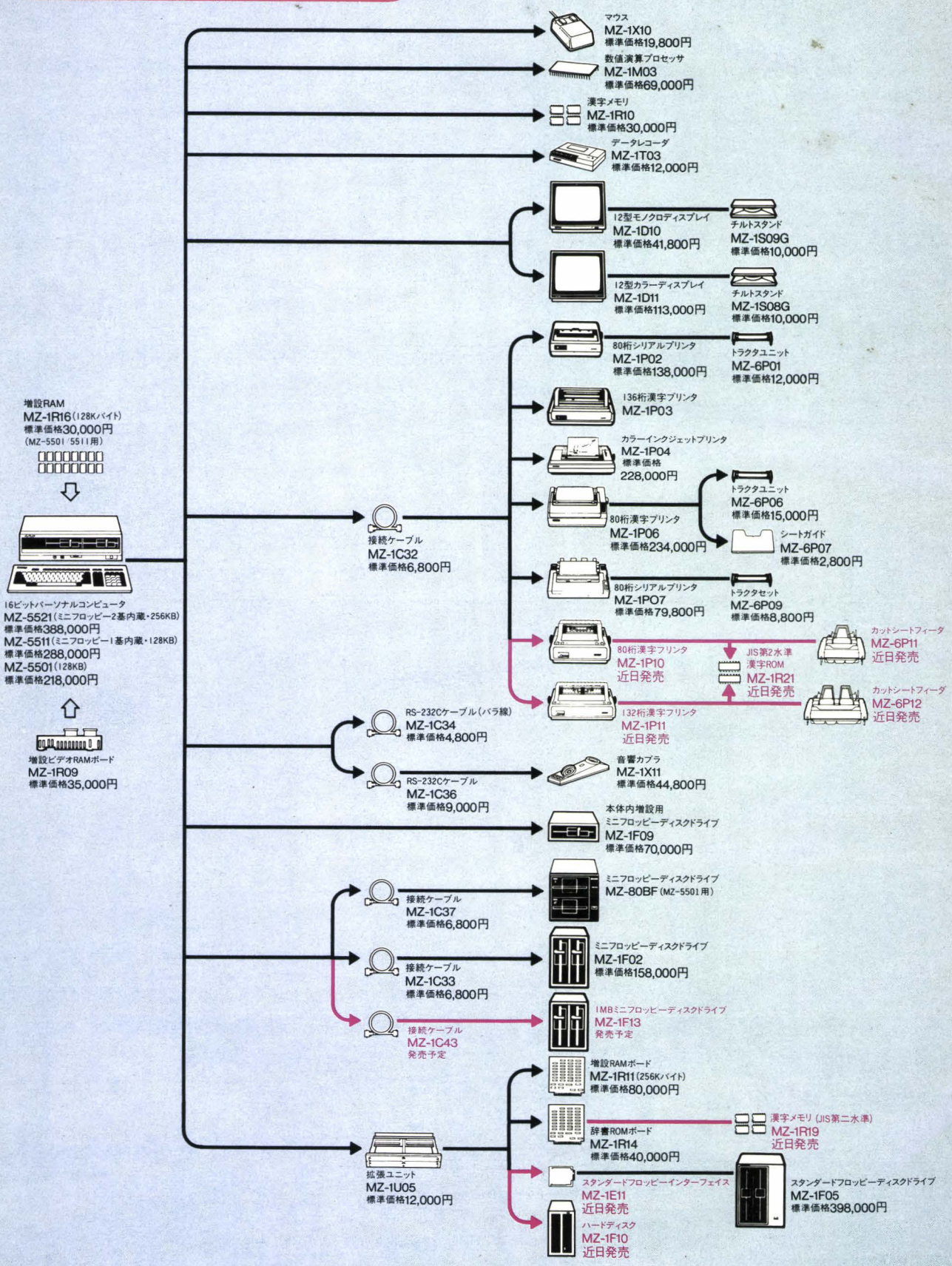
待望のディスクBASIC登場

MZ-700用ディスクBASIC
MZ-2Z009 標準価格12,000円
※ディスクBASICを使用するには、拡張ユニットMZ-1U03(標準価格 35,000円)およびミニフロッピーディスクドライブ MZ-1F07(標準価格158,000円)が必要です。

SHARP

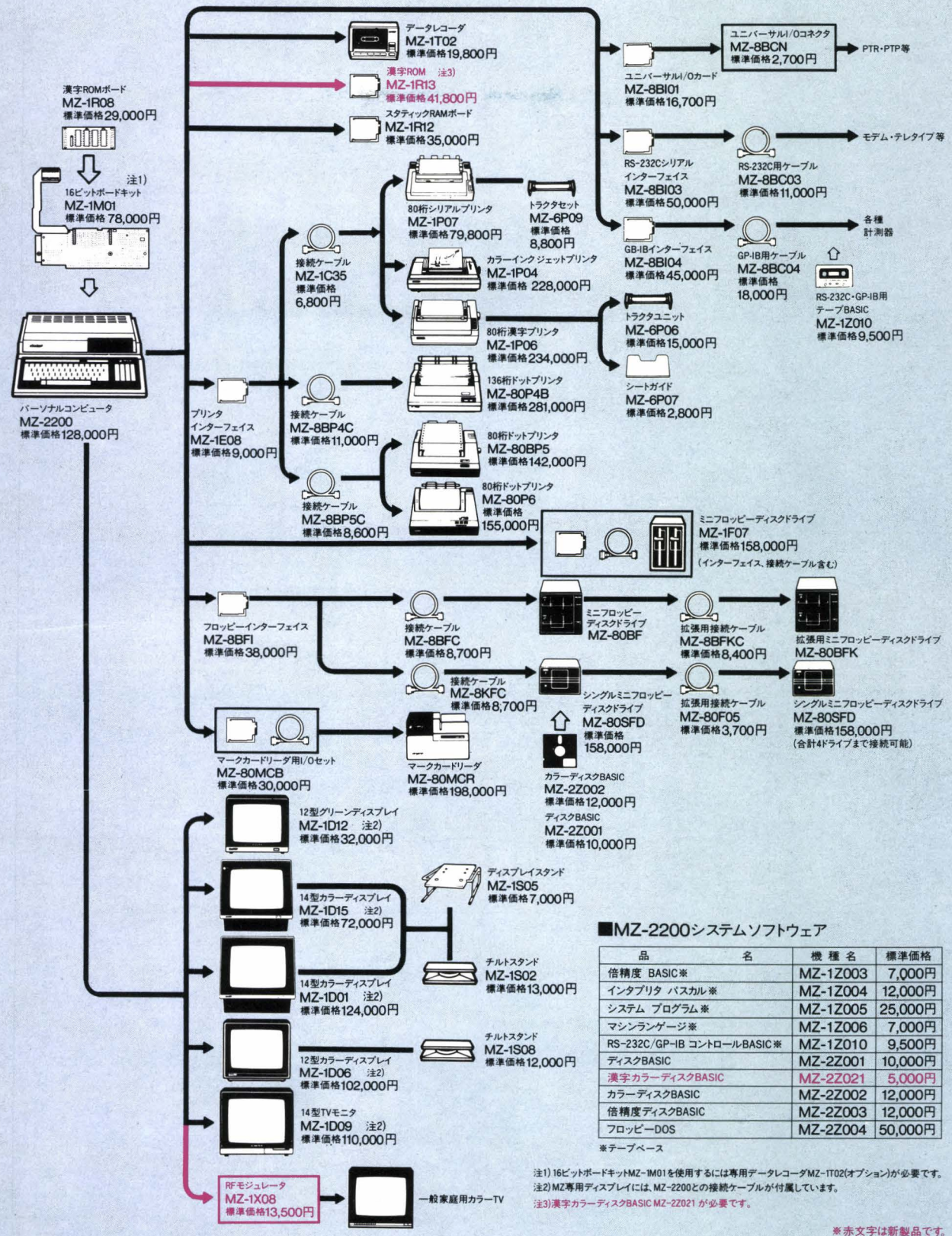
豊富なオプション機器でシステムアップ自在、充実

●MZ-5500シリーズシステム構成図



したサポート体制で優れた拡張性を誇るMZ。

●MZ-2200システム構成図



■MZ-2200システムソフトウェア

品名	機種名	標準価格
倍精度 BASIC※	MZ-1Z003	7,000円
インタプリタ パスカル※	MZ-1Z004	12,000円
システム プログラム※	MZ-1Z005	25,000円
マシンランゲージ※	MZ-1Z006	7,000円
RS-232C/GP-IB コントロールBASIC※	MZ-1Z010	9,500円
ディスクBASIC	MZ-2Z001	10,000円
漢字カラーディスクBASIC	MZ-2Z021	5,000円
カラーディスクBASIC	MZ-2Z002	12,000円
倍精度ディスクBASIC	MZ-2Z003	12,000円
フロッピーDOS	MZ-2Z004	50,000円

※テープベース

注1) 16ビットボードキットMZ-1M01を使用するには専用データレコーダMZ-1T02(オプション)が必要です。
注2) MZ専用ディスプレイには、MZ-2200との接続ケーブルが付属しています。
注3) 漢字カラーディスクBASIC MZ-2Z021 が必要です。

※赤文字は新製品です。

※漢字プリンタのキャラクタはJIS規格に準じていますので、パソコン 固有の異質なキャラクタは印字できないものがあります。

SHARP

いよいよキラめいてきたぞ、グラフィック能力。

コンピュータと映像機器とのクロスオーバー、あのスーパーインポーズ機能で、世界に先駆けて映像表現に新領域をひらいたパソコンテレビX1。まさにニューメディア時代を迎えようとしている今、そのアビリティは一段と輝きを増してきたようです。デジタルテロッパー(CZ-8DT・オプション)をはじめ、開発当初からVTRやビデオディスクシステムとのインターフェイスを考慮した先進設計。ビデオカメラで撮った思い出深い映像や、ビデオディスクの迫力ある映像に、X1で創ったコンピュータ画像を合成してオリジナル映像ファイルを創り出す、など自在の映像遊戯はもうプロの領域。もちろん、パソコンとしての高性能も見逃がせない特長です。とくにグラフィック機能やワードマスター™をバンドルしたCP/M®の登場で実務領域も一気に拡大。いま超えてるパソコンとして、ファンの熱い視線を一身に浴びています。

ビジネス分野でもX1のグラフィック能力はフルに活かされています。

●左上の表示例は近畿コンピュータサービス「実戦QC管理」より

スーパーインポーズされた画像をそのまま録画できるデジタルテロッパー。

3系統のビデオ入力端子を装備したデジタルテロッパーを接続すれば、複数台のディスプレイに映像を表示したり、映像切換えのコンソールとしても使用できます。またタイトルや図形が簡単に作成できるソフトウェアSEDSも付属しています。

デジタルテロッパーCZ-8DT 標準価格 89,800円

〈X1だからCP/M®もひと味違います〉FORTRANやCOBOLなどの高級言語を使った豊富なソフトウェアが利用できることはもちろん、X1の特長をそのまま活かせるように、パレット、プライオリティなどの特殊機能もサポート。さらに英文ワープロにもなり、やさしく親しみやすいエディターにもなるワードマスター™もバンドルする充実ぶりです。**●CZ-5CPM**(ミニフロッピーディスク用)標準価格16,800円 **CZ-3CPM**(コンパクトフロッピーディスク用)標準価格16,800円 ※CP/Mは米デジタルリサーチ社の登録商標です。※ワードマスターは米マイクロプロ社の登録商標です。

豊富なX1のソフトが共通して使えるフルコンパチブル設計

パソコンテレビ X1 シリーズ

〈X1シリーズの主な特長〉●メインCPUにZ80A(4MHz)、周辺コントロール用として2個のサブCPU搭載●RAM:メインメモリ64Kバイト、テキスト用V-RAM4Kバイト、ユーザー定義キャラクターゼネレータ6Kバイト、グラフィック用V-RAM48Kバイト(X1はオプション)●カラーBASICインタプリタ装備●セントロニクス準拠プリンタインターフェイス、ジョイスティックインターフェイス、フロッピーディスクインターフェイス(X1Dのみ)内蔵●8オクターブ3和音のサウンドゼネレータ●日付・時刻表示、番組予約が可能なカレンダー付タイマー**〈ディスプレイテレビ〉**●小さな文字や細かな縞模様もくっきりと再現する新開発ファインピッチブラウン管●RGB接続端子とビデオ入力・テレビ出力端子付。

高速・大量データ処理を実現。3インチコンパクトフロッピーディスクドライブ搭載。グラフィックRAM内蔵。14型中精細度カラーディスプレイテレビ採用。

プロフェッショナルタイプ X1D

パーソナルコンピュータ+キーボード
CZ-802C 標準価格198,000円
14型カラーディスプレイテレビ(マスクピッチ0.4㎜)
CZ-802D 標準価格128,000円
●X1Dにはローズレッド、メタリックシルバーの2色があります。

高速カセット内蔵、サブCPU搭載独立キーボードでスピーディな処理能力。パーソナルコンピュータに新領域をひらいたベストセラー機。

マニアタイプ X1

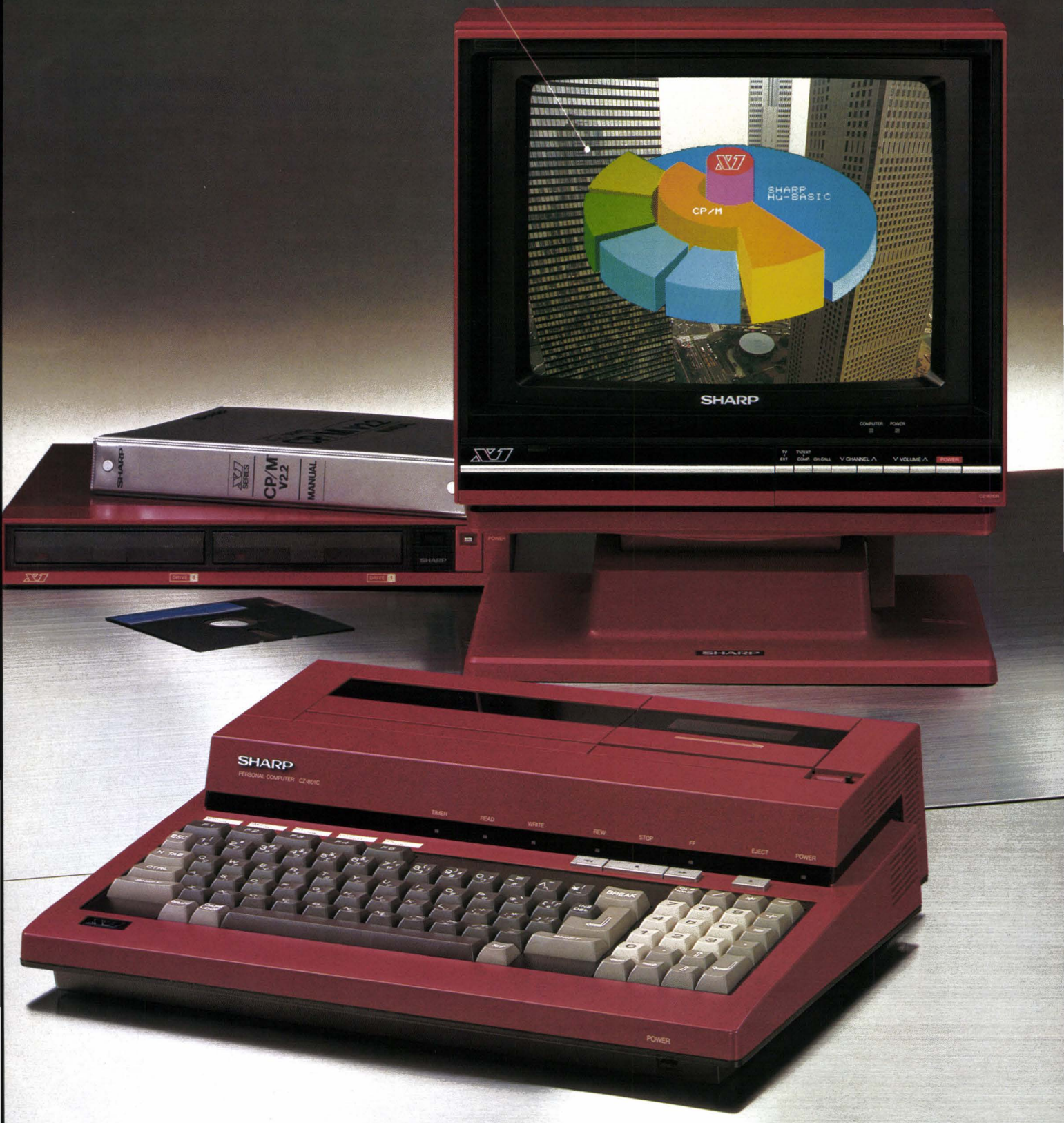
パーソナルコンピュータ+キーボード
CZ-800C 標準価格155,000円
14型カラーディスプレイテレビ(マスクピッチ0.5㎜)
CZ-800D 標準価格113,000円
●X1にはローズレッド、メタリックシルバー、スノーホワイトの3色があります。

資料請求券 X1 oh!MZ・5月号

シャープ株式会社 本社 〒545 大阪市阿倍野区長池町22番22号☎(06)621-1221(大代表)
●お問い合わせ、資料請求は…本社内電子機器事業本部システム機器営業部
またはシャープエンジニアリング㈱ 〒114 東京都北区東田端2丁目13番17号☎(03)893-4649
およびシャープビジネス㈱ 〒162 東京都新宿区市谷八幡町8番地☎(03)260-1161(大代表)

これがスーパーインポーズだ。

コンピュータ画像とテレビ・ビデオ映像を
自在に重ねあわせることができます。



使いやすさとコンパクト化を追求、キーボード、CPUを一体化。グラフィックRAM内蔵、さらにカラープロッタプリンタも装着可能。

アクティブタイプ **X1C**

パーソナルコンピュータ CZ-801C 標準価格 119,800円/14型カラーディスプレイテレビ(マスクピッチ0.45㎜)CZ-801D 標準価格 99,800円
▶チルトスタンドCZ-81TR(標準価格8,500円)はオプションです。▶写真のミニフロッピーディスクドライブCZ-801FR(標準価格198,000円)およびミニフロッピーディスク用CP/M CZ-5CPMはオプションです。●X1Cにはローズレッド、メタリックシルバーの2色があります。■ディスプレイテレビCZ-800D、801D、802DはX1シリーズのすべてのコンピュータに組合せ可能。※画面はハメコミ合成写真です。

SilverBall MZ-2000 2200 X1 COLOR USA版

パッケージのサイズが大きくなり、
遊び方の説明もくわしくなりました。

「トリをとれるは
これっきゃいねェ。」

インテリ雀士
真打登場!

パソコンに麻雀ゲームの数ある中で、グラフィックスとスピードと、あの北野英明先生のイラストレーションとで抜きん出た、SilverBall「インテリ雀士・四人麻雀」ほど真打の芸を楽しめるものはない、と自信をもって新発売。

「クイタン・サキヅケアリ」とかルールを決めて、好きなモードでゲームが出来ます。チィ・ポン・カン・ロンが可能な場合はゲームの進行が止まり、考える間がありますから初心者の独習用にも使えます。

インテリ雀士 四人麻雀

知らぬ間にウデをあげ、実戦で連勝また連勝も可能ですよ。

カセットテープ版
MZ-2000/2200用
パソコンテレビX1用
定価各3,500円
買うならこれっきゃない!

ベストセラーを続ける面白アメリカンゲーム 好評発売中!

※お求めはお近くのマイコンショップでお願いします
通信販売は一切やっていません。

★ミッションZ2000 ★オメガトロン ★フライボーイ ★ドーンパトロール ★アンダーグランドシティ ★バンパーカーズ ★ユーフォー ★レーザータンク ★サイコキューブ ★ドクタークワァティ

MZ-2000／2200用カセットテープ
定価各2,800円
このソフトはコンパイラでかかれています。

Silver Ball(商標)は、次々と新しいゲームを発表・発売していきます。ご期待ください。

製造・発売元
SilverBall
シルバーボール ソフトウェア
2855, Telegraph Ave. Suite 216, Berkeley
CA94705 U.S.A. Phone(415)848-6666

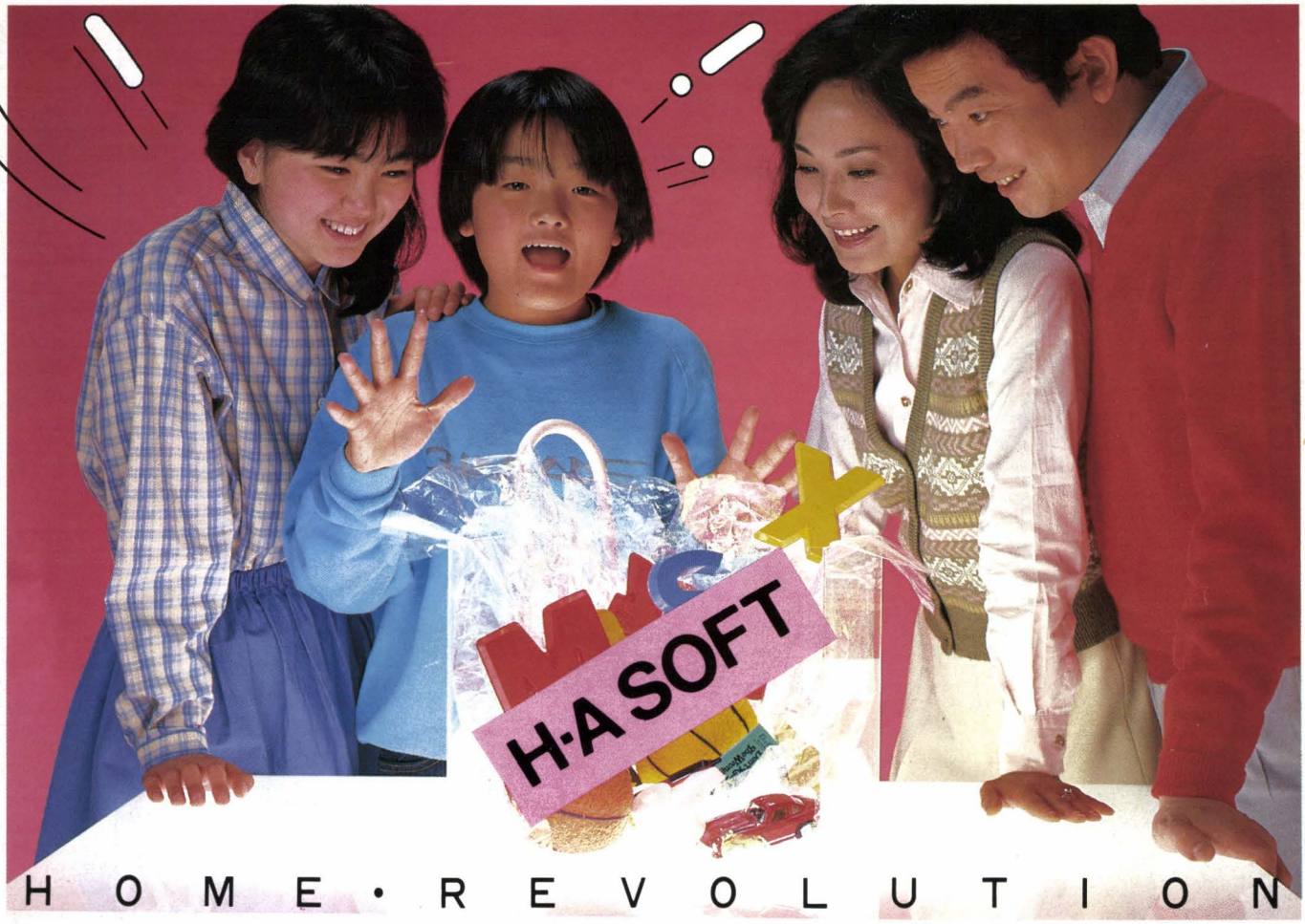
日本国内でのお問い合せは
〒102 東京都千代田区九段南2-3-14
靖国九段南ビル 電話03(263)3904
株式会社 日本ソフトバンク本社内

このテープよりの複製及び賃貸行為を禁じます。

クリエイティブ・サピエンス

家・庭・人のパソコン革命。

遊んでばかりじゃ、ただのゲーム機。
できるソフトで、H・Aスタート。



X1シリーズ用

日本語ワードプロセッサ

ユーカラJJ

漢字プリンタ不要で、本格機能

●かな漢字変換・ローマ字漢字変換、2つの入力方式が選べます。
●操作は簡単で、しかも高速処理だから、どなたでも手軽に利用できます。
●ディスク版・テープ版共にJIS第一水準漢字をサポートしました。

上位機種用ワープロに劣らない多彩な機能

■連続変換――テープ版では一文字単位、ディスク版では単語単位で連続変換ができます。ディスク版には16,400語の単語辞書。
■下線・罫線――カーソル移動で容易に下線・罫線が引いたり消したり可能
■印刷――印刷時に文書全体のレイアウトが表示され、縦書き印刷も可能。
■その他、外字・熟語の登録(ディスク版のみ)など多彩な機能満載です。

X-1用・ユーカラJJ
5"ディスク版・3"ディスク版 ¥19,800
テープ版 ¥9,800

ユーカラJJ文例集……¥3,000

簡易表計算プログラム

S・CALGO

エスカルゴ

2タイプの表で、幅広い用途

●日常の事務作業にともなう表集計が、だれでも簡単にできます。プログラミングの知識はまったく必要ありません。
●2タイプの表で用途は多彩。家計簿、カロリー計算などのホームユースから、在庫管理、現金出納帳などのビジネスユースまで、幅広く活用できます。
●指定した計算を、ファンクションキー一発で表全体に対して行います。
●ソート、検索、さらには表の中に項目の区切りなしに一行まとめてデータを入力するフリーライン入力機能など、充実したファンクション群。

MZ-700用S・CALGOもあります。

X-1用 S・CALGO　3"ディスク版…¥19,800　テープ版…¥9,800
(5"ディスク版ご希望の方には3"版と交換いたします。)

MZ-700用　S・CALGO(テープ版　¥9,800)
S・CALGOオプションパック(テープ版¥9,800)

CTC 東海クリエイト
Computer Technical Creation
〒108 東京都港区芝5-13-18 MTCビル6F ☎03-456-4615
宮崎営業所/〒880 宮崎市高千穂通1-6-35住友生命ビル7F ☎0985-29-1966

インフォメーションセンター TEL03-456-4610
●当社ソフトについてのお問合せは、上記までお気軽にどうぞ

資料請求券
MZ-5HA

WORD PROCESSOR

SHARP パソコンテレビ X1

YOU HAVE GOT TO HIRE THE NEW BRAIN OF COMPUTER AS WRITING FUTURE IN YOUR HAND.

筆啓人

STRATFORD COMPUTER CENT

X1 X1C X1D

- かな漢字変換
- かな熟語変換
- 熟語登録
- 行の挿入・削除
- 罫線・作表・アンダーライン
- コピー・転写
- 印刷範囲指定
- 行の桁数設定
- レイアウト画面表示
- タブ機能
- 各種文字のキー定義
- 倍角文字出力
- 要漢字ROM

ドットプリンタCZ-800P、漢字プリンタCZ-80PK共に使用可能／製品保証書付
フロッピーディスク3枚（システム・辞書・文例集）＋取扱い説明書40P
（高級バインダー入り）

38,000円
（5インチFD版・3インチCF版）

ラブレターから
ビジネスまで
文例集付

資料請求券
一筆啓上
oh!MZ 5

ストラットフォード・コンピューターセンター株式会社 〒336 浦和市南浦和2-36-15 ☎0488(85)5222(大代表)

★購入を希望なさる場合は上記のプログラム名・機種・電話番号・5インチ版・3インチ版の別を明記の上、現金書留でストラットフォード・コンピューターセンター Oh!MZ 係までご注文下さい。（郵送料は不要です。）

★当社ソフトウェア総合カタログをご希望なさる場合は、住所・氏名・年齢・職業を明記の上、資料請求券と切手200円分を同封して当社カタログ係までご請求下さい。

Ektachrome TRANSPARENCY
PROCESSED BY LOYAL COLOR
黄金の墓
1等賞のソフトだよ！
TITLE
カセット版 4,800円
FM-7
PC-6001 / PC-6001 mk-II
新発売 MSX(32K) ¥4800
X-07(¥4,200)
PC-6001 / PC-6001 mk-II
PC-8801
3.5インチCF版
PC-6601 ¥6,200
3.5インチCF版 ¥5,800
SMC-70/777
新発売
3インチ版
X-1/D
¥6,200
X-1/C.D用 アドベンチャーゲーム3作品完成！
FRONT
ピラミッドの謎
X-1 (要V-RAM)
X-1/D.C
ムー大陸の謎
黄金の墓
新発売
PROCESSED BY LOYAL COLOR
MSX(32K)カセット
ムー大陸の謎
X1 (カセット)
FM-7.8 (カセット・ディスクFD)
おもわずニヤッ!!
アジカルズウのおもしろソフト!!
グラフィック アドベンチャー ゲームシリーズ
カセット版 4,800円
3インチFD版 5,800円
3インチCF版 6,200円
新発売 3.5インチ SMC 70/777 6,200円
rotring
3インチFD版 9,800円
カセット版 5,800円
3インチCF版 9,800円
アイコン アニメーター
X-1/D.C (要V-RAM)
これ最高～！
X-1
JR-200
MZ-2000
2200
80B
ひらがな百人一首
カセット版 4,500円
マシン語 高速ゲーム
TANK Top
PC-8801
TANK BATA BATA
PC-8801・8001 mk-II
PC-8001 (N-BASIC)
タートル
PC-6001・PC-6001 mk-II
PC-8001 mk-II・FM-7.8
X-1
Japanese雀球
PC-8801・8001 mk-II
PC-8001 (N-BASIC)
カセット版
2,800円
マジカルズウ
Magical Zoo
ストラットフォード・コンピューターセンター株式会社 ホビー事業部
〒336 浦和市南浦和2-36-15 ☎0488(85)5222(大代表)
できたてのホヤホヤ新登場！
カセット版 3,000円
近くのお店で手に入らない人は、ちょっとめんどくさいけど プログラム名、カセットおよびディスクの別、住所、氏名、電話番号を明記の上、現金書留でストラットフォード Oh! MZ係までご注文下さい。(郵送料タダ!) 機種名も忘れずにね。
☆カタログが欲しい人は、切手200円分と資料請求券を送って下さい。☆スタッフ募集中! 詳しくは電話で
資料請求券 Oh! MZ 200

150枚を越える、ときめきの

サラダの国

ファンタジック・アドベンチャー

胸踊る数々の冒険、そしてキュートな野菜の主人公が織りなすさすらいのロマン。人気絶頂の竹・中コンビが、壮大なアドベンチャー・ストーリーを自信をもってキミに送る。

ハラハラドキドキ ストーリー

重い年貢をとりたてて国民を苦しめるカボチャ大王。彼に捕えられた反乱軍のリーダー、トマト姫。さて、絶体絶命のピンチの最中にさっそうと現れたキュウリ戦士は、トマト姫を無事救出できるか。反乱軍を勝利に導けるか。

適応機種

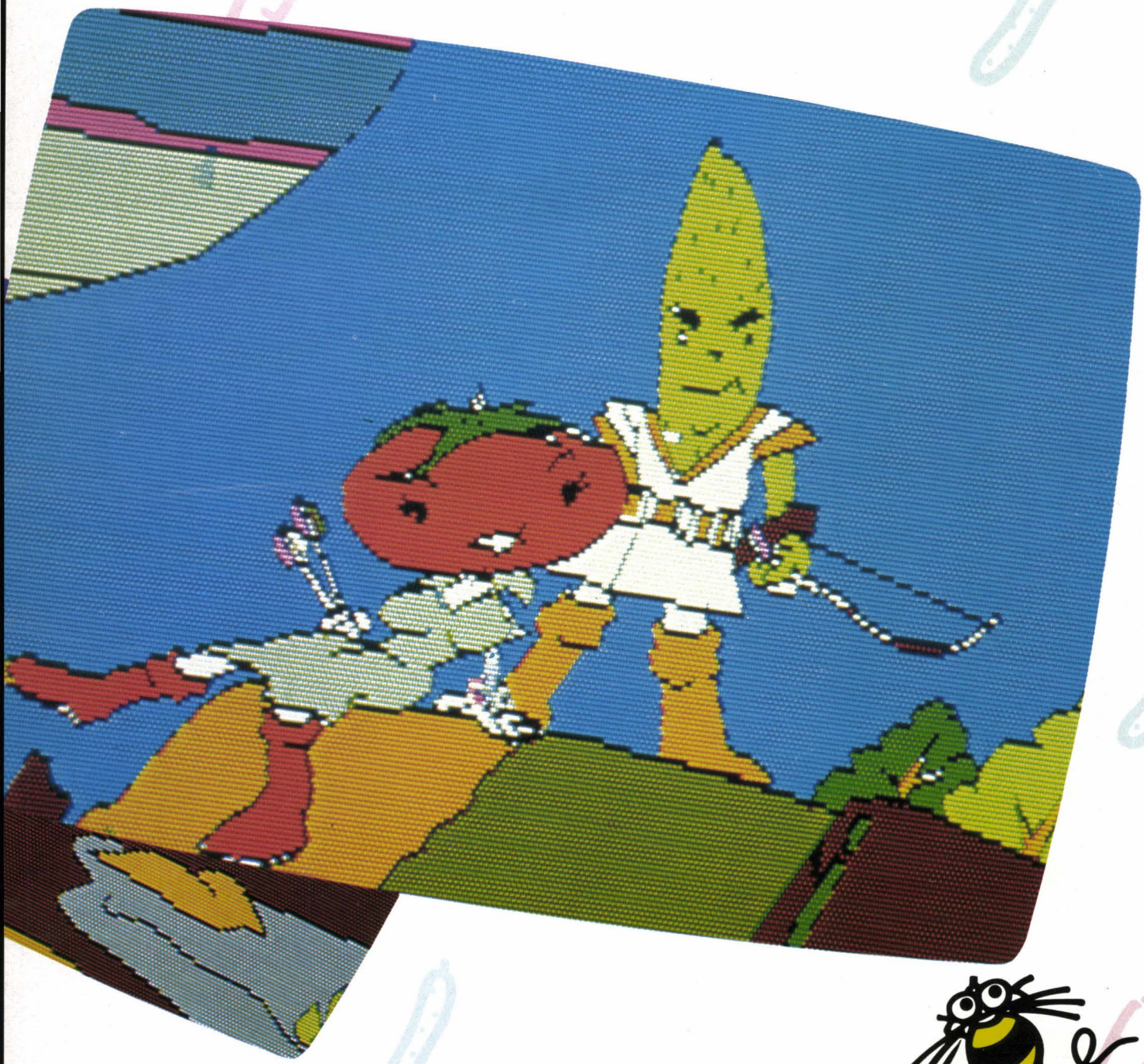
X-1D	3″FD
X-1シリーズ	T
FM-7	T
	5″FD
PC-8801/MKII	T
	5″FD
PC-9801/F	5″FD

T ¥4800　FD ¥6800

完全マシン語

ファンタジック・ピクチャー。

のトマト姫



ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II 201 PHONE:011-821-1538
ハドソン仙台/〒980 宮城県仙台市宮町1丁目4番28号 PHONE:0222-65-7031
ハドソン金沢/〒920 石川県金沢市本町2丁目1番28号 PHONE:0762-23-1263
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソン沖縄/〒901-21 沖縄県浦添市大平8番10号 PHONE:0988-79-3361

HUDSON GROUP
HUDSON SOFT®

熱戦ラリー、手に汗にぎるソフト50種。

君のパソコンに、日本ソフトバンクが自信をもっておすすめします。

倉庫番
●MZ-2000●シンキング・ラビット●T
¥3,400

内藤国雄の詰将棋
●X-I●アポロテクニカ●T/B
¥3,800

S-CALGO
●MZ-700●東海クリエイト●T/B
¥9,800

3Dゴルフシミュレーション
●X-I●T & Eソフト●T/B
¥4,000

HuWP(漢字)
●X-I●ハドソン●ROM
¥24,800

グラフィック・エディター
●XI●T&Eソフト●T/B
¥4,000

スターストリーム
●X-I●ハドソン●T/B
¥3,200

MZ-LOGO
●MZ-2000●ユニソン・ワールド●T/B
¥9,800

王　将
●MZ-2200●S・P・S●T/B
¥4,000

dB-BASIC(Ver1.2)
●MZ-2000●デービーソフト●T/B
¥5,000

ウルトラ4人麻雀
●X-I●ツクモ●T
¥3,800

アイスブロック
●MZ-2200●デービーソフト●T/B
¥3,000

ピラミッドの謎
●X-I●マジカルズウ●T/B
¥4,800

スターフリート
●MZ-2000●テクノソフト●T/B
¥4,500

倉庫番
●X-I●シンキング・ラビット●T
¥3,400

聖なる剣
●MZ-700●クリスタルソフト●T/B
¥4,200

ドンキーゴリラ
●MZ-700●テクノソフト●T
¥3,300

バグファイヤー
●MZ-700●キャリーラボ●T
¥3,000

タンク・タンク
●MZ-700●テクノソフト●T
¥3,300

モンスターハウス
●MZ-2000●日本ファルコム●T/B
¥4,500

アンダーグラウンドシティ
●MZ-2000●シルバーボール●T
¥2,800

めだかの兄弟
●MZ-2200●ハドソン●T/B
¥3,000

4人麻雀
●MZ-2000●テクノソフト●T
¥4,000

Hu-CAL II
●MZ-2200●ハドソン●T/B
¥12,800

PP-5
(ヒロトンウォーズ)
●MZ-2000●キャリーラボ●T/B
¥3,500

UFO
●MZ-2000●シルバーボール●T
¥2,800

スターストリーム
●MZ-2200●ハドソン●T/B
¥3,200

バンパーカーズ
●MZ-2000●シルバーボール●T
¥2,800

ゴルフアイランド
●MZ-2000●テクノソフト●T
¥4,000

シルクロード
●MZ-2200●ハドソン●T/B
¥3,800

ミステリーハウス Vol. II
●MZ-2000●マイクロキャビン●T/B
¥3,800

スターフリート
●X1●テクノソフト●T/B
¥4,500

コスモクロス
●X1●クリスタルソフト●T/B
¥4,800

パックマン
●MZ-2000●ツクモ●T
¥3,000

ゼクサス光速2000光年
●X1●デービーソフト●T
¥3,000

バグファイヤー
●MZ-700●キャリーラボ●T
¥3,000

カーチェイサー
●MZ-700●チャンピオンソフト●T
¥3,500

PP-13 (ウルトラセブンブリッジ)
●MZ-2000●キャリーラボ●T/B
¥3,500

王　将
●X1●S・P・S●T/B
¥4,000

王妃の首飾り
●MZ-2200●フィルコム●T/B
¥3,600

コンピュータ麻雀
●MZ-700●タスクフォーツ高知●T
¥3,500

ディメンジョナルウォーズ
●MZ-2200●ハドソン●T
¥3,200

オセロ仲間
●MZ-700●ベーシック・システム●T
¥3,000

レイザータンク
●MZ-2000●シルバーボール●T
¥2,800

カラー WICS 2000 (V2.0)
●MZ-2000●キャリーラボ●T
¥12,000

スーパーベースボール
●MZ-700●タイニーソフト●T
¥3,800

PP-9 (スターライト・アドベンチャー)
●MZ-2200●キャリーラボ●T/B
¥3,500

Data Base
●X-1●ハドソン●T
¥3,800

囲碁道場
●MZ-2000●チャンピオンソフト●T/B
¥3,800

new VIP
●MZ-700●デービーソフト●T/B
¥12,800

SOFT BANK パソコン文化のクリエイター
株式会社日本ソフトバンク

本　　社：〒102東京都千代田区九段南2-3-14靖国九段南ビル　TEL(03)263-3690代
大 阪 支 店：〒542大阪市南区難波千日前5-19河原センタービル　TEL(06)644-0191代
札幌営業所：〒064札幌市中央区南8条西3丁目10セントラルパーク83 403号　TEL(011)521-6621
仙台営業所：〒980仙台市片平1-3-2ドミー向城705　TEL(0222)67-6215
名古屋営業所：〒460名古屋市中区丸の内2丁目18番32号カーサー丸の内1102号　TEL(052)203-1838
広島営業所：〒730広島市南区東荒神町3-35広島オフィスセンター3F　TEL(082)263-8683
福岡営業所：〒812福岡市博多区千代2-21-13柴藤ビル503号　TEL(092)631-0667

特別加盟店
現在市販されているソフトのほとんどが手に入ります。

ソフトバンクにご連絡ください あなたのお近くのソフトバンク加盟店をご紹介します

●J&Pテクノランド　〒556大阪市浪速区日本橋5-6-7　TEL06-644-1413〈担当・勝田店長〉
●J&Pメディアランド　〒556大阪市浪速区日本橋5-9-11　TEL06-644-1613〈担当・山中店長〉
●J&P京都寺町店　〒600京都市下京区寺町通仏光寺下る恵美須町549　TEL075-341-3571〈担当・諸岡店長〉
●J&P渋谷店　〒150東京都渋谷区道玄坂2-28-4　TEL03-496-4141〈担当・海老名店長〉
●J&P八王子店　〒192東京都八王子市旭町1番1号八王子そごう7F　TEL0426-26-4141〈担当・河本店長〉
●SEIYU情報雑貨館マイコンショップcomica西荻窪店
〒167東京都杉並区西荻南3-14-7　TEL03-334-5311〈担当・弘中・山口〉
●マイコンショップCSK(西口)　〒160新宿区西新宿1-12-8(勧業角丸証券隣)　TEL03-342-1901〈担当・前田店長〉
●西武百貨店池袋店マイコンショップ　〒171豊島区南池袋1-28-1　TEL03-981-0111内線2928〈担当・池田、鈴木〉
●YDKシステムセンター　〒160新宿区西新宿2-4-1新宿NSビル1F　TEL03-342-2441〈担当・小川店長〉
●第1家電OAセンター　〒101千代田区神田佐久間町1-15　TEL03-253-0808〈担当・飯田〉
●ラオックス新宿店マイコンシティ　〒160新宿区新宿3-15-16ラオックスビル5F　TEL03-350-1241〈担当・菅原店長〉
●コンピュータハウスヤマギワテクニカ　〒101千代田区外神田4-3-1　TEL03-253-0121代〈担当・原〉
●大塚OAセンター八重洲　〒104中央区八重洲2-1-5東京駅前ビル2　TEL03-281-4711〈担当・金子〉
●ラオックスコンピュータメディア　〒101千代田区外神田1-13-3　TEL03-253-1341　〈担当・菅原店長〉
●J&P町田店　〒194町田市森野1-39-16 大室土地第1ビル1F　TEL0427-33-1313〈担当・河本店長〉

NEWS

Computer Books & Package Software NO.1s

実践ゲーム制作テクニック・ノウハウ集

MIAテクニカル・ライブラリ2 [書籍&カセットテープ]

PC-8801

近日発売!!

マシン語 高速ゲームの制作

A5判 定価3500円(〒300円)

マシン語ゲームの制作上、最も重要な画面処理。グラフィック・パターンのつくり方・表示・移動等の基礎から、重ね合わせの処理・仮想VRAMといった実践テクニックまで、豊富な実例(プログラム)を交じえながら平易に解説。付属のカセットには本書に掲載のパターン・エディタ、ゲーム等を収録。

STEP1／画面処理の基礎…グラフィックVRAM使用時の基本事項を解説。さらにグラフィック・パターンのつくり方、表示方法。
STEP2／画面処理の応用…リアルタイム・キー入力によるグラフィック・パターンの移動、背景と移動パターンの重ね合わせの処理、仮想VRAMといった実践テクニックを解説。
STEP3／ランダム・ノウハウ…画面処理以外のゲームづくりに役立つテクニック、ノウハウ。
STEP4／ゲーム・プログラム。

アスキー編〔全リスト公開!!〕

AXシリーズ 1500円 (〒250円)

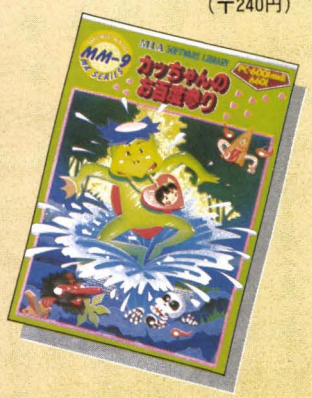
●宇宙輸送船ノストロモ●オリオン●ポリス&ギャング●スペースエネミー●パワードナイト●ザ・アステロディアン●ギャラクシーファイト●シャット・ザ・ボックス他

MIA SOFTWARE LIBRARY

MXシリーズ
MIAソフトウェア・セレクション

カッちゃんのお百度参り

PC-6601 PC-6001mkII 3500円 (〒240円)



水辺に顔を出したカッちゃんの前にカワイイ少女が。「人間になりたい」の一心でお百度参りを決意したが参道にはオバケがウヨウヨ、形も絶えず変化する難所であった。

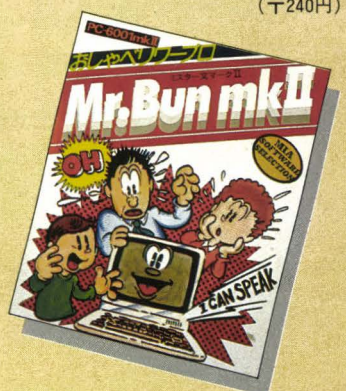
ディスク・エディタ

PC-8801 3500円 (〒240円)

5・8インチのディスク及びメモリ上のセクタ単位(256バイト)のデータを自由に変更・修正。ディスク↔ディスク、メモリ上のデータもディスクに書き込み。

おしゃべりワープロ Mr.Bun mkII

PC-6001mkII 12800円 (〒240円)



表示画面／20字×10行　漢字数／1945字　入力方式／かな漢字変換　編集機能／挿入、削除、単語登録、タブ設定、外字入力　プリンタ／NEC PC-6021、PC-8821/22、EPSON MP-80、RP-80他

資料請求券 5 MIA

■お求めは最寄りのマイコン・ショップへ。または郵送料を添えて下記へお申し込みください。

〒102 東京都千代田区紀尾井町3-20 鶴屋ビル TEL.(03)265-2461 ㈱エム・アイ・エー

MIA MICRO INFORMATION ASSOCIATES

お待ちかね！ PC-8201読本ついに登場！
特徴，活用そしてプログラムライブラリ！
8201ユーザーに必読の書だ！
発売は4月下旬(予定)。

最寄りの書店に
お申し込みください。

Techknow Lib 8200

Technical know-how and program Library

B5判 256ページ
定価1,800円

主な内容

第1部　Techknow編

ＰＣ-8201をハード，ソフト両面から分析し，ＰＣ-8201を有効に使うためのテクニカル・ノウハウをお教えします。

- 第1章　ＰＣ-8201の特徴
  - 仕様
  - メモリ構成
- 第2章　N82-BASICの内部構造
  - メモリマップ
  - RAMファイル
  - 中間言語
  - 変数領域
  - 画面
  - キーetc.
- 第3章　ＰＣ-8201の使い方あれこれ

第2部　プログラムライブラリ編

第1部のノウハウを生かして，すぐに役立つプログラムを集めました。

ゲーム，ユーティリティ，ホビー，実務など，幅広く使えます。各プログラムにはくわしい解説が付いていますので，プログラム技法にも役立てていただけます。

● 編 者 ●
㈱システムソフト

● 発 行 ●
㈱日本ソフトバンク

おまたせしました、X1ユーザーのみなさん

X1テクニカルマスター

Oh! X1!?

好評発売中!!

B5判・カラー4頁・本文295頁 定価2,500円

X1シリーズ
(X1・X1C・X1D)の
機能を100%引き出そう!!

X1シリーズが
親友になるぞぉー!

X1シリーズの多彩な機能を存分に発揮させるために開発された「SHARP HuBASIC」を平易に解説。はじめてコンピュータにふれるという方から、ある程度BASICを知っている方までを対象に、X1シリーズ(X1・X1C・X1D)入門から、簡単なマシン語までを実際の例を挙げながら、少しずつ確実に理解が深まるよう構成されています。

内容

1：X1にさわってみないか
2：How toタッチX1!
3：プログラムを作ろう
4：X1おもしろグラフィックス
5：ビジネスセクレタリーX1
6：X1と遊ぼう
7：X1ミュージックハウス
8：X1機能のいろいろ
9：機械語も使えます
〈付録〉
○便利情報「らんだむめも」
○エラーメッセージ便利表
○キャラクタコード表
——各ページを10進法，2進法，16進法で併記

ストラットフォードC.C.C. 著
発行・問合せ先
㈱日本ソフトバンク出版部
☎03(261)4095

絶賛発売中

MZ-LOGO

日本ではじめてMZ-2000に搭載!

●MZ LOGOの特徴

❶パソコンで使えるプログラミング用高水準言語で，BASICと同じ会話型インタプリタです。

❷〝タートルグラフィック〟というグラフィック機能を持っています。

❸プログラムがプロシジャ構造（細かい手順に分けられた構造）になっていて，見やすくなっています。

❹同じ作業を繰り返す再帰機能をもち，またユーザーがコマンドを自由に定義できます。

❺リスト処理機能で文字も数字も同様に扱えます。

❻作図，作画，ゲーム，人工知能，データファイルなど応用範囲が広く，初心者から専門家まで使えます。

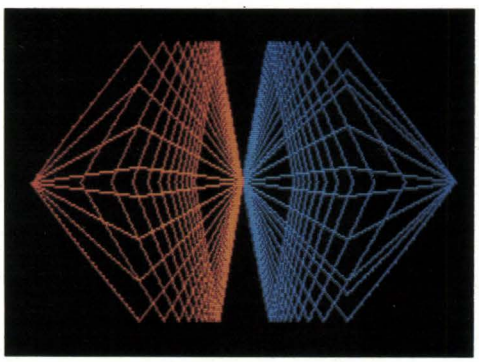
❼日本ではじめての和文LOGOマニュアル付（約100ページ）

●定価 ¥9,800（カセット版）

タートルグラフィックス　プログラム例

```
to egaku1
setpc 2
at -50 (- 50)
polyspi 100 62
end

to polyspi :side :angle
if :side < 2 [stop]
setpc pc + 1
if ((remainder pc 8) = 0) [setpc 2]
fd :side
rt :angle
polyspi :side - 1 :angle
end
```

```
to egaku2
home
rt 45
hishigata 15 0.1
lt 180
hishigata 15 0.1
setscrunch 1
end

to hishigata :n :s
if :n = 0 [stop]
repeat 4 [fd 100 rt 90]
setscrunch :s
hishigata :n - 1 :s + 0.3
end
```



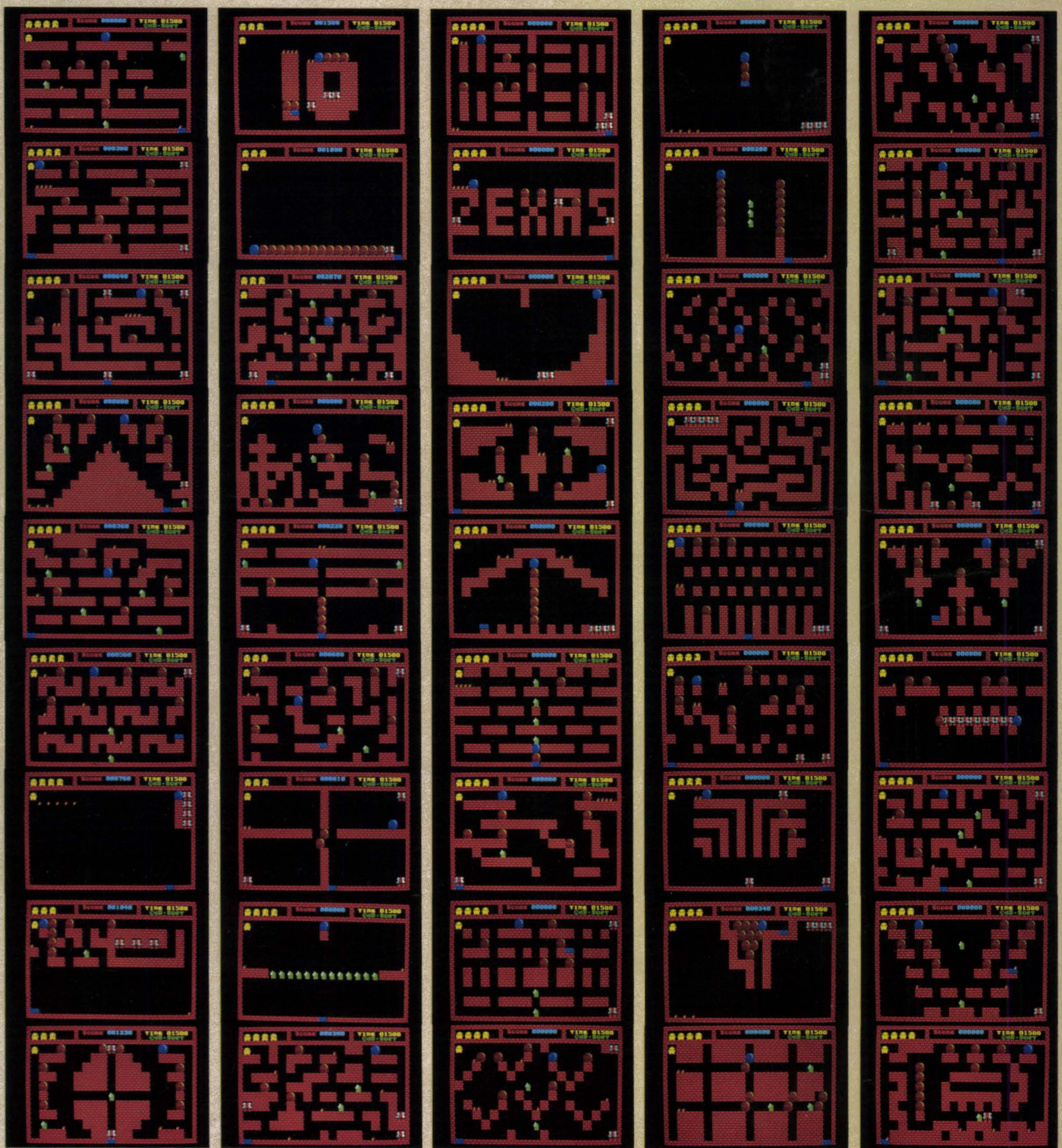
※お求めはお近くのマイコンショップでお願いします
通信販売は一切やっていません。

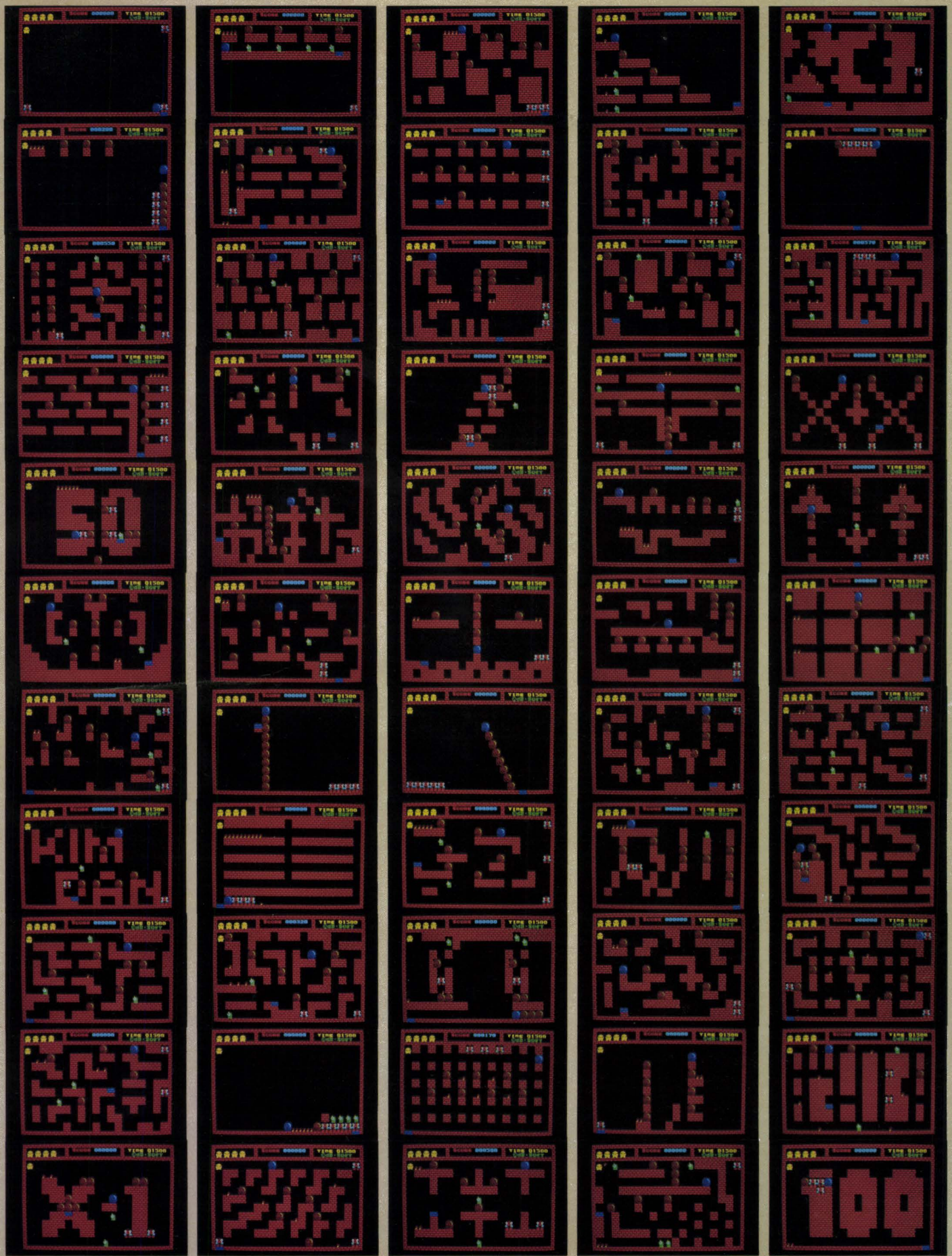
開発・発売元：UNISON WORLD INCORPORATED
2885 Telegraph Ave, Suite 216 Berkeley CA 94705 USA.

問い合わせ先：㈱日本ソフトバンクMZ LOGO係　〒102 東京都千代田区
九段南2−3−14 靖国九段南ビル　TEL 03−263−3904

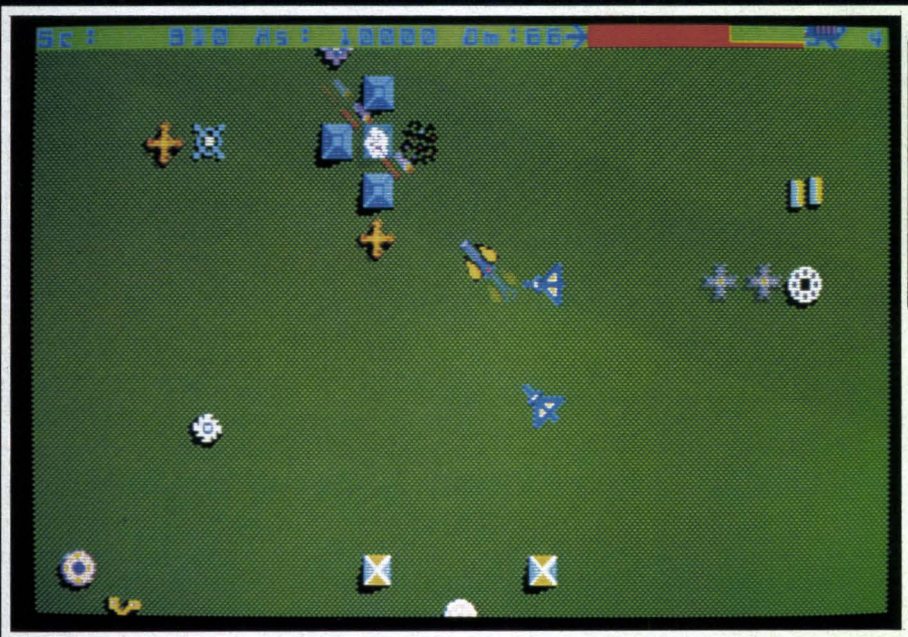
フラッピー百景

「やったね！」というには、「まだまだあるよ100画面」だけど、半分をクリアするのも、なかなかこれが至難の技。到底ここまで行けない人のため、また、行けた人には記念に、この「フラッピー百景」をプレゼントしよう。



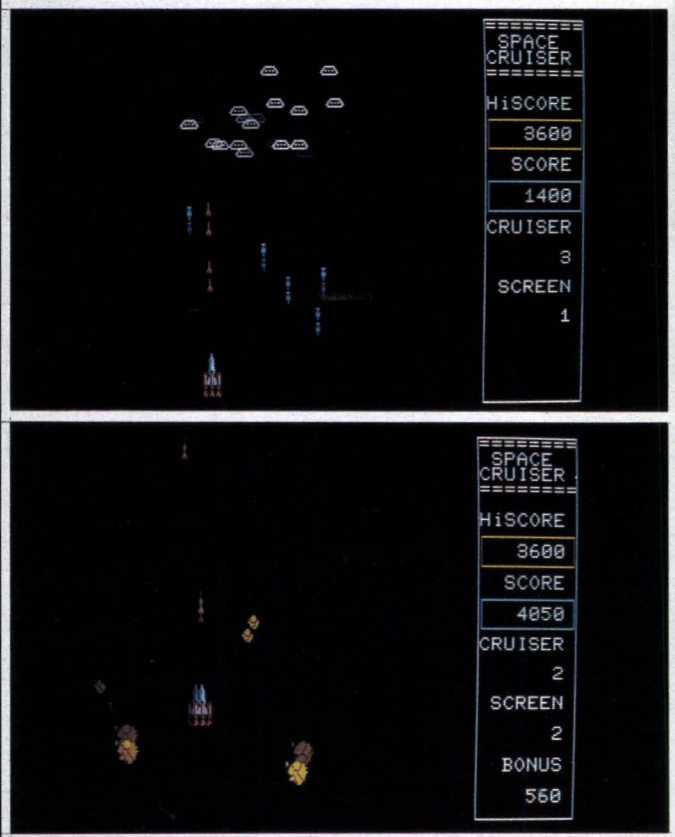


今月のソフト

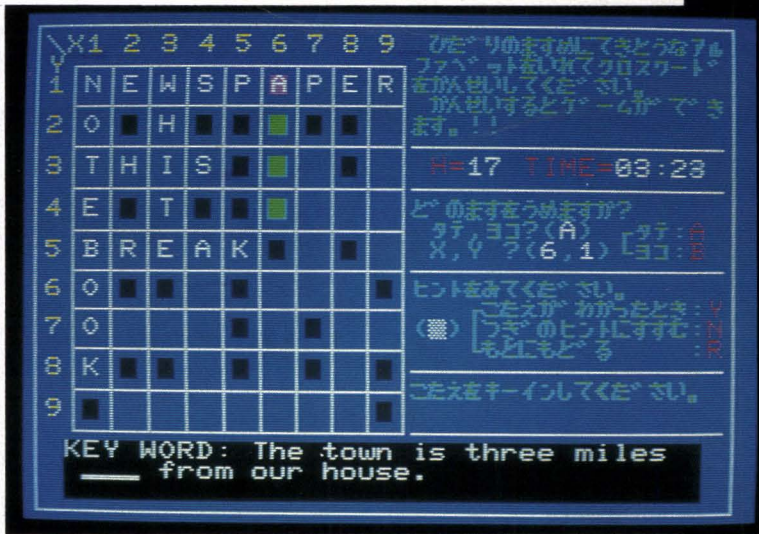


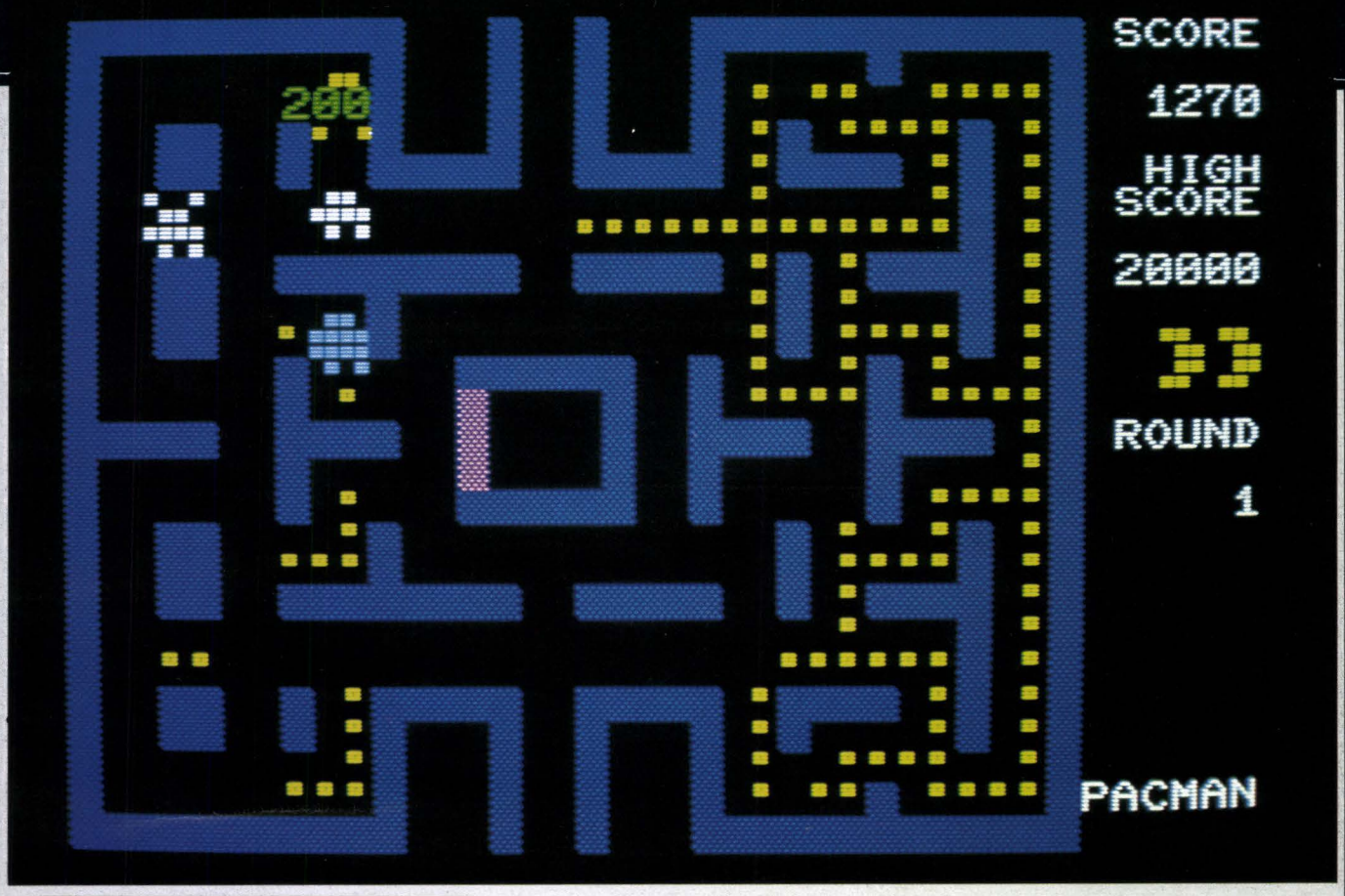
トワード8(THE SOFTOUCH 145ページ)
敵から味方に変化するキャラクタの設定やストーリー性の高いゲーム運びが評判です。

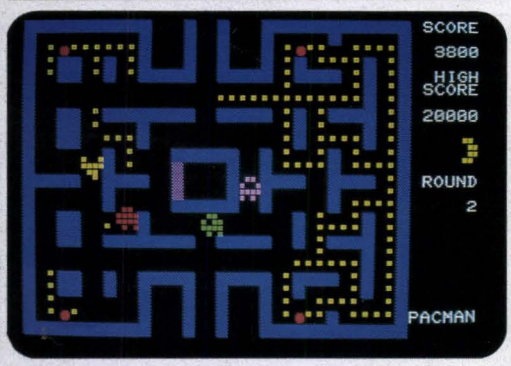
スペースクルーザー(THE SOFTOUCH 146ページ)
この流星がまた速いんですね。よけるだけでも一苦労。

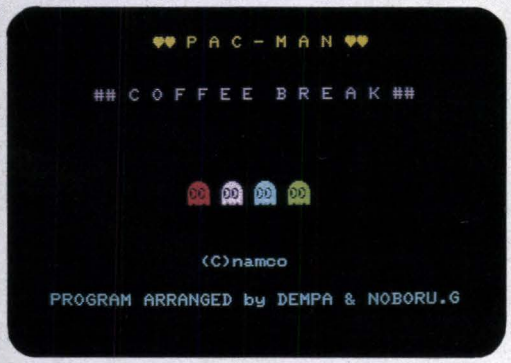


クロスワード英単語(THE SOFTOUCH 147ページ)
問題が解けるとごほうびにゲームで遊べます。

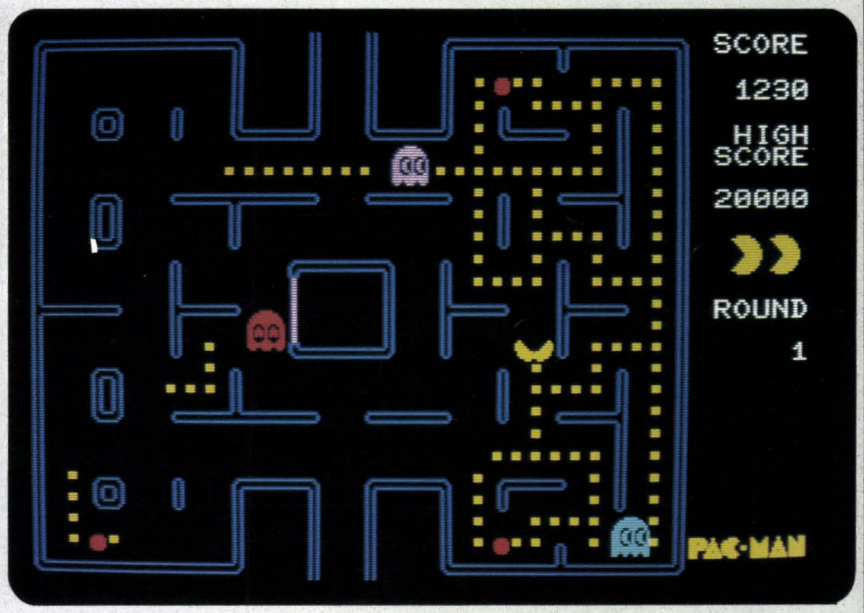


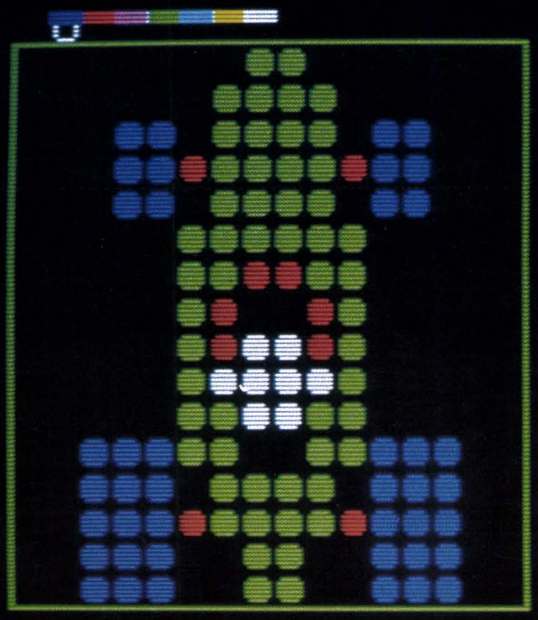






パックマン(THE SOFTOUCH 146ページ)
左と上がPCGなしの画面，左下と下がPCGを使った画面。





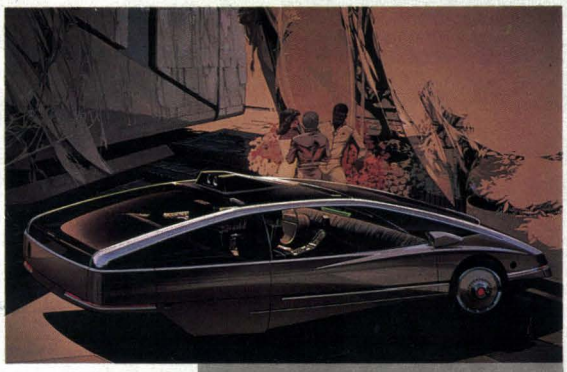
アニメディター(THE SOFTOUCH 151ページ)

インペリアル フリート(THE SOFTOUCH 150ページ)

HP王将(THE SOFTOUCH 150ページ)

Oh!MZ MAY, 1984 5

CONTENTS



Copyright Arranged by
TSURUMOTO ROOM Co.,LTD.

表紙絵：シド・ミード　SYD MEAD
*Singer's Car*より部分
表紙アートディレクション：森元　功

時速250kmで新東名高速道路を京都へ向けて〝Singer〟を走らせていた。助手席のブロンドもまぶしい秘書の黒く輝やく瞳と目線を合わせながら、京都で開かれる学会での研究発表について，最後のダメを押していた。CADに使うディスクをモニターにディスプレイして，映像とサウンドの秒きざみの進行を点検していると，リヤの様子を探っているテレスコープを通して，黒いスポーツカーの急接近を告げるシグナルがモニターに割り込んできた。路面に埋められたケーブルからの信号をコンピュータが読んでクルマは自動運航していたが，急を告げる背後に備えて，東京を出て初めてハンドルを手にし，命令を下すべくキーボードへ腕をのばした。……

1990年発刊「コンチ・オブ」より引用

〈特集〉クリーンコンピュータと言語

MZはコトバの天才である ——こうもとやすひこ 31

MZ-2200/2000(MZ-1Z004)
月間特別賞 **インタプリタパスカルの拡張** ——工藤 恵 57

ADVENTURE アドベンチャー
パッケージソフト紹介マニュアル 73
これがプロのテクニックだ —不思議の国のアリス— ——多湖明雅 90

Oh!/CZ

X1面白テクニック2 **VTR編集用ユーティリティ** ——足利裕人 122

プログラムを作るプログラム ——Elden Man 129

読み物

石のはなし **TTL物語** ——松木 透 30

パソコン千夜一夜　第5夜 **パソコン御三家と15ゲーム** ——峰岸順二 69

皿までどーぞ② **清く正しく面白く** ——祝一平 120

連載/ゲーム/ビジネス/DOS/ハード

NEWS MZ 出ましたね，DISK-700
最強無比のビジネス用BASIC 25

Z-80機械語への誘い マシン語・魔神語・DEUS EX MACHINA **ジャンプ命令** ——長瀬敏之 104

MZ-2200/2000マシン語に挑戦しよう/ **BASIC文字列サーチ・ツール** ——小笠原吉義 108

謎の円盤 CP/M® 第Ⅰ種接近遭遇 ——高野庸一 113

Oh!/ショートプログラム XIファンクションチェンジツール　MZ-2200/2000/80B
SCREEN Maker MZ-80K/C/1200 PASCAL　SP-4010用テンキー　MZ-80K/Cパターン認識プログラム 134

ポケコン横丁 **実用プログラム「ポケットメモ」** ——杉浦勇一 & 小林光二 138

カラーページ
フラッピー百景 18
今月のソフト 20
Oh!/ MZ質問箱 140
ぺんぎん情報コーナー 142
愛読者プレゼント 144
THE SOFTOUCH 145
STUDIO MZ 152
PUZZLE BOX 156
ごめんなさいのページ 157

●CP/M, CP/M-86, MP/MはDigital Research社
●MS-DOSはMicrosoft社
●FLEXはTSC社
●UCSD p-Systemは カルフォルニア大学理事会
●RACET NECDOSはRACET COMPUTES社
●SB-80, SB-86はLIFEBOAT ASSOCIATES
●WORDSTAR, MAILMERGE, SPELLSTAR, WORDMASTER, CALCSTAR, DATASTAR, SUPERSORT, INFOSTARはMicro-Pro社の各メーカーの登録商標です。
その他プログラム名，システム名，CPU名は一般に各開発メーカーの登録商標です。
本文中では"TM","®"マークは明記していません。
●本誌掲載のプログラムは，個人で利用するほかは著作権法上，無断複製を禁じられています。

熱い期待にこたえて Oh!700 スペシャル!

Oh!MZ 別冊

ADVANCED MZ-700

A4変型　約208ページ　**予価:1,800円**　㈱日本ソフトバンク出版部発行　電話03(261)4095

700ユーザーの熱い期待にこたえてOh!MZが贈るMZ-700のハイ-テック活用法
子供から大人まで楽しめるゲームなど豊富なアプリケーションプログラムを詳しく解説
初心者もベテランもこの一冊でMZ-700の多彩な機能を150%引き出せます

主な内容

1.ハイテックBASIC
- プログラミングとデバッグ
- 移植の考え方
- ゲームプログラミング
- ビジネスプログラミング

2.BASICからマシン語へ
- マシン語をあつかう法
- ディスアセンブラの作成とその活用
- エディタアセンブラの作成と活用

3.マシン語の活用とBASICの拡張
- マルチスクリーン/ハードコピー/トレーサ/ボーレート変更

4.ソフトウエアによるパワーアップ
- モニタ作成プログラム
- プリンタコンソールプログラム
- その他

5.ハードウエアによるパワーアップ
- ディスク活用法/プリンタ活用法
- PSGを作る

6.アプリケーションプログラム
- 日本語ワードプロセッサ/LISP
- 簡易経理/集計表
- 彩色四人マージャン/アドベンチャー　他

遅れて申しわけありません。
だんぜん充実して
近日刊行の予定です。

出ましたね，DISK-700

—体験レポート—

出ましたよ，FD-700

よしお　ついにシャープからMZ-700用のフロッピーディスクが，発売になったね。

ふつお　フロッピーって，LOADやSAVEが，すごく速いんでしょ。便利になるだろうね。

わるお　700もまだ見捨てられたわけじゃなかったんだな。

よしお　BASICは，シャープ仕様だけど，大幅に拡張されているんだ。なにしろ，ファイルディスクリプタが使えるし，CHAIN文も，もちろん可能だよ。

わるお　そんなの当たりまえじゃないか。

よしお　まだまだあるよ。SWAP命令で，ディスク上にあるプログラムをサブルーチンコールできるんだ。この命令を使えばどんな長いプログラムも構造化可能というわけさ。

ふつお　なんだか難しそうだけど，今までのS-BASICとは全然違ってるの？

よしお　そんなことは，ないよ。基本的な命令はほとんど同じで，そこにディスク用の命令が加わっただけと思えばいいさ。ただLOAD, SAVE時に，かならずファイル名がいるということを覚えておかなきゃだめだよ。

わるお　K/C用やテープ版S-BASIC上のソフトは，一体どうなるんだい。

よしお　心配御無用。"TRANS"っていう強力なファイル変換用のユーティリティソフトが付いていて，K/C用DISK BASICはもちろん，倍精度BASICなどの上で走るソフトは，すべてテープからもディスクからも吸い上げることが可能なんだ。それから，聞いて驚くな。なんとK/C用F-DOS上の，ソフトもコンバートするんだぞ。

わるお　なかなかやるな。ところで，ディスク以外の周辺機器のサポートはやっているのかい。

よしお　よくぞ聞いてくれましたな。プロッタプリンタのコントロールは，当たりまえだけど，そのうえ，発売予定のシリアルカードのサポートも完璧さ。きちんとシリアル用のファイルをオープンすることができて，リストやデータを簡単に送れるんだ。データしか送れない中途半端なソフトとは，わけが違うよ。

ふつお　シリアルカードって，何のことなの？

よしお　RS-232Cという通信用インタフェイスのカードのことさ。これを使えば他のコンピュータと通信できるんだ。音響カプラを併用すれば，「ウォーゲーム」みたいに電話回線を通して遠方のコンピュータをアクセスすることも夢じゃないよ。

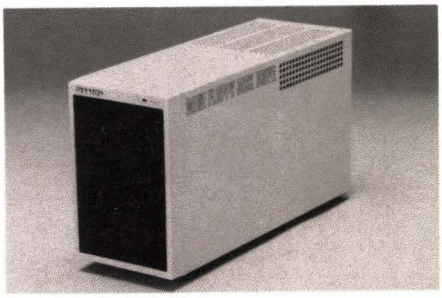
わるお　ウーン！　こいつはすごいや。さすがはシャープだな。早速注文しようっと。

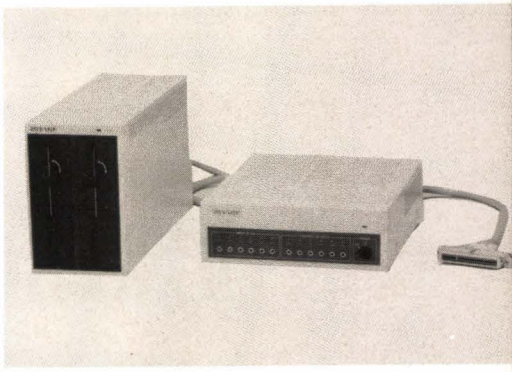
よしお　最初は，文句ばっかり言ってたくせに……。

わるお　どうせ私は，ドジでまぬけなマイコン少年ですよーだ。

ふつお　ねえねえ，Oh！ MZの編集室にある700用フロッピーを使ってみようよ。話ばかりじゃよくわかんないよ。

わるお　そうだそうだ。えーっと，必要な





装置は……。

ふつお　接続の仕方が複雑だな。本体から拡張ユニットへつながって，それからフロッピーへ行くんだね。

よしお　そうそう。全部の装置に電源をつなぐことを忘れないようにね。

わるお　インタフェイスカードにIPLローダーROMを差し込むことも絶対に忘れちゃだめだぞ。

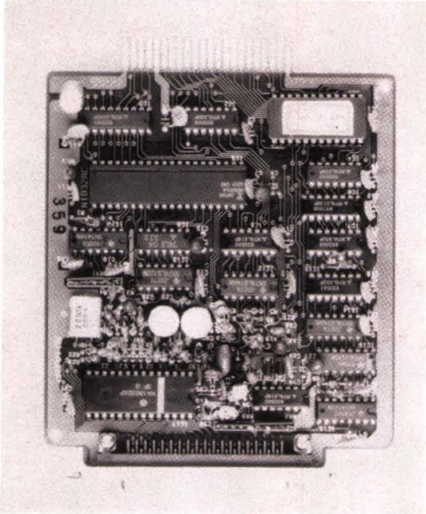
ふつお　ROMって何なの？

わるお　プログラムが書き込んであるICのことさ。これがないとDISK BASICをブート（注：IPL-LOADのこと）できないんだ。MZ-700の場合F000Hからのメモリ空間にあって，電源ONのあと，Fコマンドで DISK BASICが走るんだ（しかしMZ-1F07用のインタフェイスカードにもROM用ソケットがあるなんて驚きだな。わっ，オフレコ，オフレコ！）。

よしお　ディスケットの差し込み方は，「親指の法則」でやればいいんだ。つまり，ラベルの貼ってある面を親指で触るように，かつ，ラベルが手前にくるように持ってドライブに差し込んでくれたまえ。

ふつお　右側のドライブ1に入れるんだね。早速何かプログラムを作ってみようよ。

よしお　待って待って。その前にマスター



ディスクのバックアップをとらないとだめだよ！

わるお　バックアップ用のユーティリティも付いているんだな。えーっと。

RUN"FD COPY"

として新しいディスケットをフォーマットしてから，マスターディスクのコピーをつくればいいんだろ。

DISK BASICの特徴

……とまあ何とか３人は，ディスクを使い始めたようです。それではここで私の独断と偏見にもとづいてディスク用に追加されたコマンド類のやさしい説明をいたしましょう(隠しコマンドもあります)。

ファイル名について

700のDISK BASICでは，デバイスの指定(どこにSAVE,LOADするかということ)をファイルディスクリプタで行います。

FD１:，FD２:，CMT:，RS１:，RS２:，などが可能です(カセットは，CAS:ではありません。注意してください。また，RSは，シリアル(RS-232C)用です。さらにデータファイル用としては，CRT:，LPT:も使えます)。ファイル名は14文字までで，エクステンションは使えません。BSD, BTXなどのファイル属性は700側で判断してつけてくれます。 ↗

NEWS MZ

最強無比のビジネス用BASIC

MZ-5500用のBASICとして，シャープ㈱からBASIC-３(MZ-2Z017)が発売されました。店頭でご覧になった方も多いでしょうが，ヒューレットパッカード系の巨大BASICで，すぐれた操作性，豊富かつ強力なコマンド体系など，これまでのBASICの概念を超えたすぐれものです。

第一水準の漢字メモリ(MZ-1R10)，シャープオリジナルの辞書ROMボード(MZ-1R14)を用意すれば，ローマ字入力，文節変換によって日本語処理ができるのは，BASIC-2と同じですが，ビジネスアプリケーションには不可欠の，表作成コマンド(TABLE)，検索・並べ換えに威力を発揮するSEARCH，MAXS，MINS，プログラムを読みやすくするためには仮名８文字までの変数やGOTO"漢字ラベル"を使えるし，なによりも独立ローカル変数が使えるのがうれしいかぎりです（わたしの知るかぎりローカル変数の使えるBASICは，OS-9にあるのみです)。

DATA入力，DATA表示，DATA保護もBASICコマンドとしては万全になっており，内部演算は10進処理で変換誤差をもちません。

シャープではこのBASICをビジネス用のメインにしたい考えで，マニュアルもＢ５判400ページを越える詳細なもので，「入門編」，「基本編」，「応用編」，「文法編」，「付録」の５編によって初心者への配慮がうかがえます。ビジネス用とはいえ，SOUND, MUSIC, COLORもサポートしており，ファイル処理にすぐれていますから，楽しみの多いBASICと言えます。

BASIC-3は漢字CP／M-86(2Z-017)とセットで20,000円です。

〔特長サンプル1〕

```
100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 !!                                                                     !!
120 !!      有効桁数は １２桁 で－９９～＋９９までの指数表示              !!
130 !!                                                                     !!
140 !!            内部演算は １０進演算 で変換誤差が無い。                 !!
150 !!                                                                     !!
160 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
1000 A=1:B=1:Z=0.1
1010 DISP TAB (1);"A+B";TAB (20);"A-B";TAB (41);"A*B";TAB (62);"A/B"
1020 FOR I=1 TO 22:C=A+B:D=A-B:E=A*B:F=A/B
1030 DISP C;TAB (19);D;TAB (40);E;TAB (61);F:B=B*Z
1040 NEXT I
1050 END


RUN
 A+B                A-B                  A*B                  A/B
 2                  0                    1                    1
 1.1                0.9                  0.1                  10
 1.01               0.99                 0.01                 100
 1.001              0.999                0.001                1000
 1.0001             0.9999               0.0001               10000
 1.00001            0.99999              0.00001              100000
 1.000001           0.999999             0.000001             1000000
 1.0000001          0.9999999            0.0000001            10000000
 1.00000001         0.99999999           0.00000001           100000000
 1.000000001        0.999999999          0.000000001          1000000000
 1.0000000001       0.9999999999         0.0000000001         10000000000
 1.00000000001      0.99999999999        0.00000000001        100000000000
 1                  1                    1E-12                1E+12
 1                  1                    1E-13                1E+13
 1                  1                    1E-14                1E+14
 1                  1                    1E-15                1E+15
 1                  1                    1E-16                1E+16
 1                  1                    1E-17                1E+17
 1                  1                    1E-18                1E+18
 1                  1                    1E-19                1E+19
 1                  1                    1E-20                1E+20
 1                  1                    1E-21                1E+21
Ready
```

↘
例　SAVE"FD 1:LUM"
　　(ドライブ1にLUMという名前のファイルをSAVEする)。
　　LOAD"CMT:KYOKO", A
　　(カセットから，KYOKOという名前のアスキー形式のファイルをLOADする)。

注意：オプションとしてコンマAを付けるとアスキー形式でSAVE, LOADが可能です。ただしオプションは，LOAD時にも必要です。

ユーティリティTRANSについて

かなり強力なファイルコンバータです。K/C用(F-DOS上のファイルを含む)のソフトを，テープからもディスクからも読み込みます（**わるお**　80B用F-DOSからも読み込めるぞ。SOURCEの指定時に5を押すんだ。だけど700と80Bでは，アスキーコードが違うから文字が化けるぞ)。

注意：F-DOS用BASIC COMPILERのソースリストには，行番号が付いていないので，いったん文字列データファイルとしてオープンして変数に読み込ませ，行番号を付けて別のファイルを作り，再びLOAD"ファイル名", Aとする必要があります。

ファイルAUTO RUNについて

ブート直後に，自動的に"AUTO RUN"(真ん中のスペースをお忘れなく)が走ります。購入時には，ファンクションキーの定義を行っています。私は，COLOR,,4,0を追加して気色悪い青いバックカラーから抜けています。

ファイルスペック(FILESPEC)とは

"ファイルディスクリプタ：ファイルネーム"を合わせて，あるいはその一部をファイルスペックと言います。CP/Mなどの場合はさらにファイルタイプ（エクステンションともいう）を指定できますが，このDISK BASICではBASICが勝手につけてくれます(OBJ, BTX, BSD, BRDなど)。

I)DIR"ファイルディスクリプタ"

〔特長サンプル2〕

```
100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 !!                                                          !!
120 !!              ====== "TABLE" ======                       !!
130 !!                                                          !!
140 !!              作表コマンドと2種類の罫線                   !!
150 !!                                                          !!
160 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
200 FOR I=1 TO 9:DISP :NEXT I
201 !                       ── 疑似罫線（キャラクタ罫線）使用
202 !                       ── X軸の開始座標
203 !                       ── X軸方向の罫線の間隔
204 !                       ── Y軸の開始座標
205 !                       ── Y軸方向の罫線の間隔
1000 TABLE /,5,8,7,7,7,7;18,2,2,2:CURSOR 19,16:DISP "罫線－1"
1010 CURSOR 11,18:DISP "| A | B | C | D"
1020 CURSOR 5,20:DISP " 田 中 "
1030 CURSOR 5,22:DISP " 上 田 "
1040 CURSOR 5,22:DISP " 上 田 "
2000 TABLE ,45,8,5,5,5,5;18,1,1,1:CURSOR 56,16:DISP "罫線－2"
2010 CURSOR 55,18:DISP "1    2    3    4"
2020 CURSOR 46,19:DISP "TANAKA"
2030 CURSOR 46,20:DISP " UEDA ":CURSOR 0,15

RUN
```

罫線－1

	A	B	C	D
田 中				
上 田				

罫線－2

	1	2	3	4
TANAKA				
UEDA				

```
Ready
```

〔特長サンプル3〕

```
100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 !!                                                          !!
120 !!         ========= "SEARCH" =========                     !!
130 !!                                                          !!
140 !!     文字変数や文字配列変数の中から文字列を探す。         !!
150 !!                                                          !!
160 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
170 DIM A$(9)*10:N=1
180 A$(1)="ABCFNZGHIJ":A$(2)="KLSHAZQRST":A$(3)="UVWXYZABCD"
190 A$(4)="ｱｲｳｴｵｶｷｸｹｺ":A$(5)="ｻｼｽｾADﾁﾂﾃﾄ":A$(6)="ﾅAZﾈﾉﾊﾋﾌﾍﾎ"
200 A$(7)="ﾗﾘﾙﾚﾛﾜﾗAZﾝ":A$(8)="ｲAｸﾅﾆﾇﾁﾏｷﾝ":A$(9)="PSUHJﾅﾆﾇHL"
201 !                 ── 検索の対象となる総括的文字配列変数
202 !                 ── 検索を始める配列変数の順位
203 !                 ── 探す文字列（－は探す文字列の前後関係）
204 !                 ── 見付けた順位を代入する変数
205 !                    （見付から無い場合は0を代入）
1000 SEARCH A$(*),N,/"-AZ-",P:IF P=0 GO TO 9999
1001 !                ── 検索の対象となる文字配列変数
1002 !                ── 変数の中の検索を始める位置
1003 !                ── 探す文字列
1004 !                ── 見付けた位置を代入する変数
1005 !                   （見付から無い場合は0を代入）
2000 SEARCH A$(P),1,"AZ",Q:N=P+1:IF N>9 THEN 9999
3000 DISP "A$ (";P;") の";Q;"文字目から以降に あります。"
3010 GO TO 1000
9999 END

RUN
A$ ( 2 ) の 5 文字目から以降に あります。
A$ ( 6 ) の 2 文字目から以降に あります。
A$ ( 7 ) の 8 文字目から以降に あります。
Ready
```

〔特長サンプル4〕

```
100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 !!                                                            !!
120 !!         =========== "CREATE" ===========                   !!
130 !!                                                            !!
140 !!           ランダムファイルの領域を確保する。               !!
150 !!           レコード数  1～65535                             !!
160 !!           レコード長  2～65535文字まで可                   !!
170 !!                                                            !!
180 !!         (OPEN:RFORMAT:PUT:GET)                             !!
190 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
200 DIM D$(9)*10:ON ERROR GO SUB 7000:KILL "B:BRD-TEST.DAT"
210 FOR I=1 TO 9:D$(I)=STRING$ (10,RIGHT$ (STR$ (I),1)):NEXT I
220 DATA 11,12,"一",21,22,"二",31,32,"三",41,42,"四",51,52,"五"
230 DATA 61,62,"六",71,72,"七",81,82,"八",91,92,"九"
1000 CREATE "B:BRD-TEST",20,70
1001 !              ── 1レコード長70バイトで
1002 !                 （DATAブロックの最大の長さ）
1003 !              ── レコード数20個分を
1004 !                 （記録するDATAブロックの数）
1005 !              ── ファイルネーム"BRD-TEST"として
1006 !              ── ドライブBに
1007 !              ── ランダムファイルの領域を確保する。
2000 OPEN "R",1,"B:BRD-TEST"
2001 !              ── ドライブBの"BRD-TEST"を
2002 !              ── 取り扱い番号1のファイルで
2003 !              ── ランダムファイルと決めて
2004 !              ── 取り扱いを開始する
3000 RFORMAT #1,A,B,C$,D$(J)
3001 !              ── 取り扱い番号1のファイルの
3002 !              ── レコードの中のDATAの組み合わせを決める
4000 FOR J=1 TO 9
4010 READ A,B,C$
4011 !              ── 取り扱い番号1のファイルの
4012 !              ── J番目のレコードへ
4020 PUT #1,J:REM   ── A,B,C$,D$ (J) を書き込む
4021 !                 （RFORMATで決めたDATAを）
4030 NEXT J
5000 FOR J=9 TO 1 STEP -1
5001 !              ── 取り扱い番号1のファイルの
5002 !              ── J番目のレコードからDATAを
5010 GET #1,J:REM   ── 読み出し A,B,C$,D$ (J) に代入する
5011 !                 （RFORMATで決めたDATAに）
5020 DISP A,B,C$,D$(J)
5030 NEXT J
6000 CLOSE :END
7000 IF ERL=200 THEN RESUME NEXT

RUN
 91          92          九          9999999999
 81          82          八          8888888888
 71          72          七          7777777777
 61          62          六          6666666666
 51          52          五          5555555555
 41          42          四          4444444444
 31          32          三          3333333333
 21          22          二          2222222222
 11          12          一          1111111111
Ready
```

FILESではありません。あくまでDIRです。CP/Mの雰囲気を充分に味わってください。ファイルディスクリプタ(FD1:またはFD2:)を省略するとデフォルトドライブ(デフォルト文で指定されたか，あるいは直前に使用していたドライブ）の中にあるファイルを表示します。

2) DEFAULT〝ファイルディスクリプタ″

DEVICEではありません。これもあくまでDEFAULTです。LOAD, SAVEでファイルディスクリプタが省略された場合，DEFAULT文で指定された装置が対象になります。

3) SWAP〝ファイルスペック″

CHAIN文は単にディスク内のプログラムに実行を移すだけですが，SWAP文ではEND命令で再び元のプログラムに帰ってきます。つまりディスク上のプログラムをサブルーチンコールできるのです（ただしRETURNでなくENDで帰る）。これはかなり強力な命令で，プログラムの構造化に大いに役立つことと思います。

4) XOPEN　#ロジカル番号,〝ファイルスペック″

ランダムアクセスファイルを作るこの命令がDISK　BASICの最大の特徴であると言っても過言ではありません。WOPENやROPENで作ったシーケンシャルファイルと異なり，一つひとつのデータに番号（データ要素番号）がついていて，何番目のデータでもランダムに読み込み，書き込みが可能です。PRINT #ロジカル番号，(データ要素番号)，変数で書き込みます。INPUT #の場合も同様です。

ロジカル番号というのは，いくつもファイルをオープンした場合，それらを識別するためにつける番号です。1から127まで可能です（ひとつしかファイルをオープンしない場合は，1にしておいて間違いありません。また，もちろんMAXFILES文は使えません)。

〔特長サンプル5〕

```
  100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  110 !!                                                            !!
  120 !!        ========== "MAXS":"MINS" ==========                 !!
  130 !!                                                            !!
  140 !!     数値配列変数（DIM）の中の最大値:最小値をさがす。          !!
  150 !!                                                            !!
  160 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 1000 D=30:DIM A(D):RANDOM
 1010 FOR I=1 TO D:A(I)=INT (RND 100):NEXT I
 1020 MAXS A(*),1,DA
 1021 !                   最大値を探せ
 1022 !                   最小値を探せ
 1023 !                   探がす配列変数
 1024 !                   A(*)で配列変数を総括して表現出来る。
 1025 !                   探し始める位置
 1026 !                   位置を代入する変数
 1027 !                   同じ値が複数あれば添字の小さい方を代入
11030 MINS A(*),1,DI
 1040 DISP "ｻｲﾀﾞｲ ﾁ ﾊ A(";DA;")=";A(DA),
 1050 DISP "ｻｲｼｮｳ ﾁ ﾊ A(";DI;")=";A(DI):DISP
 1060 FOR I=1 TO D:DISP "A(";I;")=";A(I),:NEXT I


RUN
ｻｲﾀﾞｲ ﾁ ﾊ A( 27 )= 95                   ｻｲｼｮｳ ﾁ ﾊ A( 2 )= 0

A( 1 )= 23        A( 2 )= 0         A( 3 )= 73        A( 4 )= 33
A( 5 )= 51        A( 6 )= 73        A( 7 )= 58        A( 8 )= 17
A( 9 )= 59        A( 10 )= 67       A( 11 )= 35       A( 12 )= 38
A( 13 )= 3        A( 14 )= 47       A( 15 )= 35       A( 16 )= 58
A( 17 )= 31       A( 18 )= 59       A( 19 )= 77       A( 20 )= 41
A( 21 )= 17       A( 22 )= 24       A( 23 )= 46       A( 24 )= 12
A( 25 )= 48       A( 26 )= 84       A( 27 )= 95       A( 28 )= 5
A( 29 )= 36       A( 30 )= 74
Ready
```

〔特長サンプル6〕

```
  100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  110 !!                                                        !!
  120 !!     ======= 独立ローカル変数が使える。 =======            !!
  130 !!                                                        !!
  140 !!          メインルーチン と サブルーチンで                 !!
  150 !!          同じ変数が別のものとして扱える。                 !!
  160 !!                                                        !!
  170 !!              (CALL & GOSUB)                            !!
  180 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
  200 !----------------------- MAIN -----------------------
  201 !                独立変数が使えるサブルーチンを呼び出す
  210 A=123:REM                              サブルーチン名
  220 DISP "1)ﾒｲﾝ ﾙ-ﾁﾝ ﾉ A=";A;:CALL ｻﾌﾞ2000
  230 DISP "2)ﾒｲﾝ ﾙ-ﾁﾝ ﾉ A=";A;:GO SUB 1000
  240 DISP "3)ﾒｲﾝ ﾙ-ﾁﾝ ﾉ A=";A
  999 END
 1000 !-------------------- ｻﾌﾞ 1000 --------------------
 1010 A=789:DISP "  ｻﾌﾞ 1000 ﾉ A=";A:DISP :RETURN
 2000 !-------------------- ｻﾌﾞ 2000 --------------------
 2001 !        CALL文で呼ばれるサブルーチンである事を示す
 2002 !        サブルーチン名             メインルーチンへもどす
 2010 SUB ｻﾌﾞ2000:REM
 2020 A=456:DISP "  ｻﾌﾞ 2000 ﾉ A=";A:DISP :SUBEND


RUN
1)ﾒｲﾝ ﾙ-ﾁﾝ ﾉ A= 123   ｻﾌﾞ 2000 ﾉ A= 456

2)ﾒｲﾝ ﾙ-ﾁﾝ ﾉ A= 123   ｻﾌﾞ 1000 ﾉ A= 789

3)ﾒｲﾝ ﾙ-ﾁﾝ ﾉ A= 789
Ready
```

〔特長サンプル7〕

```
 100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 110 !!                                  !!
 120 !!   仮名を変数として使える。          !!
 130 !!                                  !!
 140 !!       ===============            !!
 150 !!                                  !!
 160 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 200 ﾔﾏﾓﾄﾏｻﾋﾛ=123
 210 ﾔﾏﾓﾄﾏｻﾋｺ=456
 220 ｼｮｳｹｲ1=2500
 230 ｼｮｳｹｲ2=35
 240 ｺﾞｳｹｲ=ｼｮｳｹｲ1+ｼｮｳｹｲ2
 300 DISP "ﾔﾏﾓﾄﾏｻﾋﾛ=";ﾔﾏﾓﾄﾏｻﾋﾛ
 310 DISP "ﾔﾏﾓﾄﾏｻﾋｺ=";ﾔﾏﾓﾄﾏｻﾋｺ
 320 DISP "    ｺﾞｳｹｲ=";ｺﾞｳｹｲ


RUN
ﾔﾏﾓﾄﾏｻﾋﾛ= 123
ﾔﾏﾓﾄﾏｻﾋｺ= 456
    ｺﾞｳｹｲ= 2535
Ready
```

〔特長サンプル7-2〕

```
 100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 110 !!                                                          !!
 120 !!    GOTO"漢字ラベル名" と GOTO（変数）が可能                  !!
 150 !!                                                          !!
 160 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 200 A=1234:GO SUB "松田聖子"
 210 A=1000:B=600:C=400:GO SUB (A+B+C)
 999 END
1000 "松田聖子"
1010 DISP A:RETURN
2000 REM
2010 DISP A+B+C:RETURN


RUN
 1234
 2000
Ready
```

〔特長サンプル8〕

```
 100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 101 !!                                                        !!
 102 !!     =============   MOVE   =============               !!
 103 !!                                                        !!
 104 !!              プログラムの増殖命令                         !!
 105 !!                                                        !!
 106 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
 107 !
 108 !                      プログラムのコピーを作る。
 109 !                      文番号100から
 110 !                      106までのプログラムのコピーを
 111 !                      文番号3000で始まり
 112 !                      増分10のプログラムとして作る。
 113 !
1000 MOVE 100,106,3000,10
1001 !                      行番号1000を消しRETURN
1002 !                      してからLISTを見て下さい。
```

わるお君の「FIELD文が使えなくてレコード長が32バイト固定じゃないか」という声が聞こえてきそうですがひたすら無視しましょう。

5)その他

LOCK, UNLOCK, KILL, MERGE, SEARCH, LABEL, EDIT, IF～THEN～ELSEなどが追加されていますが言わずもがなでしょう。

80桁ボードやCP/Mも

……と，まあ私の目についた命令だけを説明しましたが全体的に見て良くできたDISK BASICであると言って良いと思います。

ラベルが使えるなど，ディスク以外の命令も拡張されていますのでHuBASICのユーザーでも大きな不満はないと思います。(ただしコントロールキーがほとんど使えないのには閉口しますが) 最後に，3兄弟がぶつぶつひとりごとを言っていますので聞いてみましょう。

わるお　ドライブと拡張ユニット，それにマスターディスク(**IPL用ROM付属**)を合わせると20万円位かかるじゃないか。高いなあ。インタフェイス用カードだけ別売になるといいんだけど。それにHuBASICのDISK版はでないのかな。ハドソンに問い合わせてみようっと。

よしお　MZ-700用の周辺機器はこれから続々と発売されますよ。既にPCGや32K-BYTE CMOS RAMボードはでているし，シリアルカードは発売予定らしいですね。僕としては，80桁ボードやCP/Mが出て欲しいですな。

ふつお　物凄く長いアドベンチャーゲームがでないかな。アップルには，ディスク5枚組のものがあるんだって(後藤貴行)。

なお，価格は12,000円(IPL ROMを含む)で，起動するためにはディスクドライブ(MZ-1F07)および拡張ユニット(MZ-1U03)の増設が必要です。

〔特長サンプル9〕

```
100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 !!                                                                  !!
120 !!            =========== "KEYIN" ===========                       !!
130 !!          INPUTしたDATAがそろえて表示される                         !!
140 !!             入力エラーはCLKeyで自動復帰                            !!
150 !!           リターンKeyで入力前のデータが有効                         !!
160 !!                                                                  !!
170 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
1000 DIM A(12):FOR J=1 TO 10:DISP :NEXT J:TABLE ,10,8,14,8,14;15,1,1,1,1,1,1
1010 FOR I=1 TO 12:A(I)=I*100+I
1020 CURSOR 10+INT (I/7)*22,14+I-6*INT (I/7):DISP I;"ｶﾞﾂ"
1030 CURSOR 30+INT (I/7)*22,14+I-6*INT (I/7)
1040 KEYIN A(I),13,2,"F,"
1041 !                    └─3桁毎のコンマの有無
1042 !                   └──符号の表示方法:ゼロサプレス (N, F, L, R)
1043 !                 └────小数点以下の桁数
1044 !              └───────INPUTする総桁数
1045 !                         (符号:コンマ:小数点を含む)
1046 !         └────────────受け取る変数
1050 NEXT I
9999 END

RUN

   1 ｶﾞﾂ
```

〔特長サンプル9-2〕

```
100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
110 !!                                                                  !!
120 !!         ====== USING と IMAGE ======                             !!
130 !!                                                                  !!
140 !!              イメージ記号による書式指定                            !!
150 !!                                                                  !!
160 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
1000 DIM D@*20
1001 !  ┌──────────書式を指定する。
1002 !  │    ┌─────5個のスペースを空ける。
1003 !  │    │  ┌───12個の文字変数の為の空白を確保する。
1004 !  │    │  │    ┌─"の値は"と言う文字を書く。
1005 !  │    │  │    │    ┌負の時に－を表示(正の時は空白を表示)
1006 !  │    │  │    │    │ (＋なら負の時は－正の時は＋を表示)
1007 !  │    │  │    │    │ (￥又は＄ならそれぞれ￥や＄を表示)
1008 !  │    │  │    │    │ ┌数値の桁を指定する。
1009 !  │    │  │    │    │ │     (＃の所はゼロサプレスされる)
1010 IMAGE "5X 12A の値は  -#,##Z.##"
1011 !                        │  │ └─小数点以下2桁まで表示する。
1020 FOR I=1 TO 5:REM          │  └──この桁はゼロサプレスしない。
1030 READ D@,A:REM             └────三桁目にコンマを入れる。
1040 DISP USING 1010;D@;A
1041 !                 └───────変数Aを
1042 !              └──────────文字変数D@の後ろへ
1043 !         └───────────────文番号1010で指定された書式で書く。
1050 NEXT I
2000 DATA "DATA 1",1234,"DATA 2",-1.256,"DATA 3",0
2010 DATA "DATA 4",-987.1,"DATA 5",.05
9999 END

RUN
   DATA 1 の値は  1,340.00
   DATA 2 の値は     -1.25
   DATA 3 の値は      0.00
   DATA 4 の値は   -987.10
   DATA 5 の値は      0.05
Ready
```

〔特長サンプル10〕

```
100 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
101 !!                                                                  !!
102 !!        =========== 三つのプログラム連結命令 ===========             !!
103 !!                                                                  !!
104 !!           "LOADSUB":"CHAIN":"MERGE"                              !!
105 !!                                                                  !!
106 !!   デイスク上のプログラムをメモリー上のプログラムにつなぐ               !!
107 !!                                                                  !!
108 !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
109 !
110 !   ****************** LOAD SUB *****************
111 !   * デイスク上の指定したのプログラムをLOADして
112 !     メモリー上のプログラムの後ろに接続又は入れ替えをする。
113 !   * LOAD後はLOADSUB命令のすぐ後ろから実行する。
114 !   * LOADするプログラムの行番号の変更が可能。
115 !
116 !   例  LOAD SUB"B:TEST":1000,1000,10
117 !                  *          *    *   *
118 !        ドライブ名 ─┘
119 !        ファイル名 ──────┘
120 !        消去するプログラムの先頭行番号 ─┘
121 !                 (又はラベル名)
122 !        LOADするプログラムの先頭に付ける行番号 ─┘
123 !        LOADするプログラムの行番号の増分 ──────┘
124 !
125 !   ******************** CHAIN *******************
126 !   * デイスク上の指定したのプログラムをLOADして
127 !     メモリー上のプログラムの後ろに接続又は入れ替えをする。
128 !   * LOAD後の実行開始ラインの指定が可能。
129 !   * アスキーファイルのLOADも可能。
130 !
131 !   例    CHAIN / "B:TEST":1000
132 !                *  *          *
133 !        CLOSE ──┘
134 !        ドライブ名 ─┘
135 !        ファイル名 ───────┘
136 !        LOAD後実行するラインナンバー ─┘
137 !                  (又はラベル名)
138 !
139 !   ******************** MERGE *******************
140 !   * デイスク上の指定したのプログラムをLOADして
141 !     メモリー上のプログラムに重ねるようにつなぐ。
142 !   * 行番号の重なった処は読み込んだプログラムに置き換わる。
143 !
144 !   例    MERGE "B:TEST"
145 !                 *
146 !        ドライブ名 ─┘
147 !        ファイル名 ─────┘     ( *印の所は省略出来る。)
148 !
149 !   ************************************************************
150 !
151 !                                  LOAD SUB  CHAIN   MERGE
152 !   プログラム中での使用               ○       ○       ×
153 !   マニュアル使用                     ○       ○       ○
154 !   サブルーチンでの中での使用         ○       ×       ×
155 !   サブプログラム中での使用           ×       ×       ×
156 !   アスキーファイルのLOAD             ×       ○       ○
157 !   LOAD時のラインナンバー変更         ○       ×       ×
158 !.  LOAD後の実行ライン指定             ×       ○       ×
159 !   LOAD後の変数の保持                 ○       ○       ×
160 !
```

新連載

石のはなし ❶

TTL物語

Toru Matuki
松本 透

ハードウェアを構成するものの中で，もっとも基本的かつ重要な部品に「TTL」というのがあります。ここでは，「TTL」がどんなものであるか，お話ししたいと思います。

トランジスタの集合体だ！

今となっては，「真空管」を見たり触ったりしたことのある人はごくごく一部の存在になってしまったような感があるのですが(「マジックアイ」なんて覚えていますか!?)，トランジスタを知っている方は多いと思います。そう，トランジスタというのは，図1のように足（導線）が3本出ている部品(半導体)ですね。ちょっと昔，3000円くらいで買えた単3電池2本で動くAMラジオのフタを外してみると，こういうのが7個くらい入っていたはずです。

このトランジスタを記号で書くと図2のようになります。そう，シリコン（ケイ素）やゲルマニウムに，不純物を混入してN型とP型を作り，それらの組み合わせで，NPNとPNP型ができます。

ところで，トランジスタの大きさは数ミリ角で，ちょうど人の前歯と同じかもう少し小さいくらいなのですが，実際の中身は米ツブよりもはるかに小さいくらいで，あとの大部分は指でつまみやすいようにできているだけです。中身だけを10個や100個くらい並べても大した大きさになりません。これがIC（Integrated Circuit＝集積回路）の根本的な考え方なのです。

TTLはICの一例だ！

トランジスタの親類にFETという部品があります。FETというのは電界効果トランジスタの意味で，外見はトランジスタにそっくりなのですが，電気的な特性は真空管のように電圧で制御するタイプです（トランジスタは電流で制御します)。そして，FETもICにすることができます。ですから，普通のトランジスタのICはこれと区別するために，特にTTL(Transistor-Transistor Logic）と呼びます。

一方FETのうち，MOS-FETという種類は，集積度をとても高くすることができる

図1

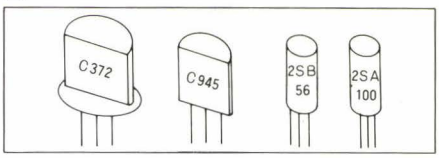
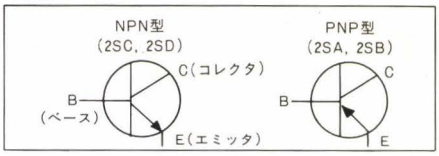


図2



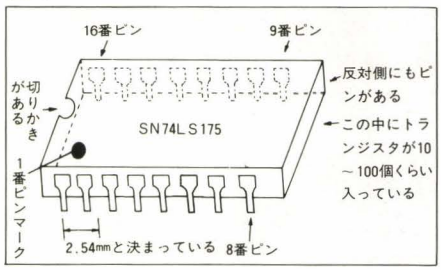
という特長があります。集積度のとても高いICは，特にLSI(大規模集積回路)と呼び，大抵MOS-FETでできていると思って間違いありません。Z-80CPU，なんていう部品も，もちろんそうです。

また，MOS-FETをコンプリメンタリという回路構成（IC内部で）にすると，極端に消費電力を少なくできるというメリットがあります。そのかわり集積度を上げるのは難かしくなるので，主にTTLのような簡単なICの置き換えとして使われることが多いようです。このようなICは，特にC-MOS（コンプリメンタリMOSの意味）と呼びます。ちなみに，Z-80などはN-MOSのLSIと呼びます。たとえば，普通のパソコンでTTLを使うような場所でも，ハンドヘルド機ならC-MOSを使うのです。電池が長持ちしますからね。

TTLを見てみよう

シャープのパソコンというのは，何かと本体を開ける機会に恵まれて（？）いて，グラフィックRAMや拡張ユニットを取り付けたりで，皆さんも一度くらいは中身を見たことがあるかと思います。もし，まだでしたら，一度マニュアルを参考に本体のフタを開いてみてください（他社のマニュアルには，こんなことは書いてないっ！)。これから買う，という人は，本誌1月号のカラーページを見ると良いでしょう。MZ-5500の中身が写真で載っていますから。

図3



さて，TTLは，2×1cmくらいの大きさで足（ピン，導線）が片側に7個～8個，両側で14個か16個ある黒い物体で，背に〝74LS×××〟などと書いてあるはずです。×××は2～3桁の数字で，先頭にSNなどがついているもの（TI社製品等）もあります。また，〝7406〟のようにLSのないもの，最後に〝N〟などのアルファベットがついているものもあります。1台のパソコンには，20～200個くらいのTTLが入っています。TTLは，米国のTI（テキサスインストゥルメンタル）社がオリジナルを数百品種出していて，日米の多くのメーカー，日本では日立，三菱，富士通などがライセンス契約をして，同じように使えるピンコンパチブルなものを発売しています。たとえば，TI社のSN74LS00と日立のHD74LS00，三菱のM74LS00，富士通のMB74LS00とは完全に交換できます。

なお，TTLと同じ形でも，4116とか4164と書いてあるものは，メモリ用LSIです。8255，8253，Z-80のような足が40個もあるものも，制御用LSIです。

つづきはまた次回，おたのしみに。

特集 クリーンコンピュータと言語

MZはコトバの天才である

Yasuhiko Koumoto
こうもとやすひこ

クリーンコンピュータは神話でも伝説でもない。MZは，80K/C/1200の伝統を確実に継承し，古典的な言語から先端的な言語，インタプリタからコンパイラまで，すべてを揃えてユーザーの〝熱中〟を待っている。コンピュータが言語で動く以上，これほどぜいたくなパーソナルコンピュータがあったろうか。今回は，BASIC以外の言語から先輩たちの遺産を特集してみた。

はじめに

MZはクリーンコンピュータです。というのはユーザーが使用できるメモリを大量に持っていますから，BASIC以外の言語を簡単に走らせることができるという特徴を持っているということです。MZ-80K/Cの時代には、その特徴が最大限に生かされ，いろいろな言語が発表されて，まさにクリーンコンピュータの黄金時代だったと言えるでしょう。しかし，現在において，それらの言語はほとんどが忘れ去られているのではないでしょうか。クリーンコンピュータで使える言語には何があるかと聞かれたとき，インタプリタPASCAL，WICS，BASEなどはすぐ思いついても，ほかはなかなか言えません。そこで，埋もれてしまった言語を発掘しようということで今回の特集が生まれたわけです。対象は，80K/Cの子孫ともいえる700です。しかも，テープ版のもののみを集めてみました。といっても，700用には2～3のシステムソフトとBASE，WICSしか発表されていません。そこで，今回紹介する言語はもともとが80K/C，1200用に発表されたものばかりになっています。しかし，それらの言語は，ほとんどがそのまま700で使えるようになっていますから心配はいりません。そのままでは動かないものでも，本誌83年4月号で発表されたモニタSP-1002の改造版であるFN-700からロードしてやればもう完全といっていいほど動かすことができます。FN-700にモニタを変換することが必要なものは，それを示してあります。

なお，今回紹介する言語は，GAME，PALL，FORM(FORTRAN-MZ)，microPASCAL，TL/1，LISP，FORTH(FORTH/Σ，M-FORTH)の7種類です。これらの中には販売されていないものもありますが，機械語の全リストが雑誌で発表されていますから，それを打ち込む気力と体力があれば使用することができるでしょう。

それでは，クリーンコンピュータ言語の世界に入っていくことにしましょう。

GAME-MZ
（モニタ変換）

概　要

GAMEはTiny BASICともいえるVTLなどの記号言語を拡張したもので，78年5月号のASCII誌で初めて発表されました。製作者の大西博氏は6800版のものを発表したわけですが，その後，8080，Z80，6502などの各CPUに移植され，ASCII誌上ではBASICと並ぶ共通言語として注目されていたようです。

GAMEはもともと整数型BASICをもとにしていますから，機能もそれをそのまま受け継いでいます。それに加えて，16進数の処理機能や間接モードによるメモリアクセス機能を持っているため，機械語的なオペレーションをするのに便利になっています。しかし，GAMEの特徴は何といってもその実行速度でしょう。これは，命令の表現が単純になっている（その反面，プログラムは読みにくい）ため構文解釈が速くできるためで，当然といえば当然ですが，速度を要するゲームなどのプログラムにはうってつけのものです。GAMEとはGeneral Algorithmic Micro Expression（一般的な算法のための小型表現とでも訳すのでしょうか）の略語だそうですが，製作者の頭の中にはこれでゲームを作ることが第一にあったのではないでしょうか。

GAMEにはまた，GAME自身で書かれたコンパイラが発表されています。GAMEインタプリタだけでもBASICの2倍の速さであるのに，コンパイラを通すと約20倍の速さになることがわかっていますから，これは驚き以外の何物でもありません。

ところで，GAMEには「超CPUシミュレーション言語」というなんとも凄いうたい文句が付けられています。これは，16ビットの汎用レジスタを26本とエディット機能を持ち，記号命令で動くような超CPUを考えるとき，GAMEはその動作のシミュレーションをやっているのにほかならないということです。これは，Tiny BASICとGAMEの違いを明確にするための理解の仕方のひとつですが，少し大袈裟なような気もします（しかし，このうたい文句は大好きです）。

エディタ・システム関係

表1.1にエディタおよびシステム関係のコマンドの一覧表を示します。また，図1.1にGAME-MZのメモリマップを示します。

GAMEにおけるプログラムの作り方はBASICと何ら変わりがありません。つまり，キーボードから，

行番号　文

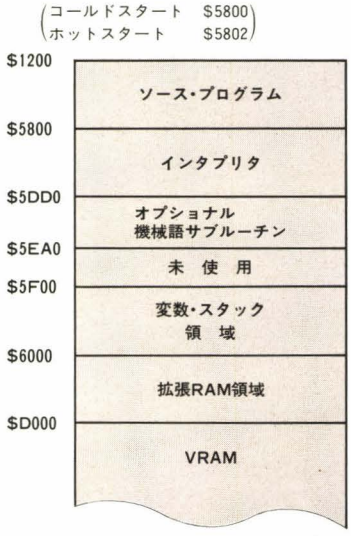
という形式でプログラムを打ち込んでいけばいいのです。もちろん，スクリーンエディットをすることも可能です。

作成したプログラムはカセットテープにセーブ，ロードをすることができます。この機能はGAMEがMZに移植される際に付け加えられた機械語サブルーチンによっています。したがって，実際にセーブロードを行なうときには，GAMEの機械語サブルーチンコールの命令でメモリの番地を直接

表1.1　GAME-MZのエディタ・システム関係のコマンド

コマンド	例	意　味
0		すべてのプログラムのリスティング。
行番号/	100/	行番号からのリスティング。
#＝n	#＝1	n以上の値をもつ最初の行番号から，プログラムを実行。
&＝0		ソーステキストのクリア。
&:0)＝$FF		ソーステキストを修正可能にする。
&:0)＝$F0		ソーステキストを修正禁止にする。
＝＝		ソーステキストの末尾を探してEOFマークをセットする。
＝n	＝$2000	ソーステキストの格納番地をnからとみる。
＊＝n	＊＝$A000	ソーステキストの格納番地の上限をnにする。
＞＝$5DD0 ␣"File Name"	＞＝$5DD0␣"P1"	RAMの内容をファイルネームを付けてカセットにセーブする。＝から＆で示される領域がセーブされる。
＞＝$5E00		テープの内容をロードする。
＞＝$5E20		テープの内容をベリファイする。

図1.1　GAME-MZメモリマップ



コールします。すなわち，セーブは，

＞＝$5DD0〝ファイルネーム〟

の形式で，そして，ロードは，

＞＝$5E00

の形式で行ないます。また，テープの内容をベリファイするには次のようにします。

＞＝$5E20

GAMEによるプログラムを実行するには，GAMEのGOTO命令(＃＝)を使って実行を開始する行番号へジャンプします。このとき指定する行番号は実際に実行する行番号より小さくて構いませんから，プログラムを最初から実行させるときは，

＃＝1

でよいのです。またプログラムの実行は，SHIFT と BREAK を同時に押すことで中断することができます。

文　法

● 定数・変数

GAMEで扱える数は10進数と16進数です。16進数は数字の前に$を付けて10進数と区別します。

(例)　1，10，$0D，$80A0

変数名はA～Zまでの26種類に限られます。ただし，冗長な名前を付けることができますから，変数にある程度意味を持たせることができます。

(例)　A，B，ACC，ASSOC

(ただし，A，ACC，ASSOCは同じものと見なされる)

● 配列

GAMEではBASICのDIM文に相当する配列宣言はなく，ユーザーがメモリ上に直接指定する方式になっています。配列へのアクセスは，配列名にメモリ上の番地を代入し，そこからの変位を指定することで行ないます。たとえば，

A＝$8000

であるとき，

A:n)

は，$8000＋n番地，

A(n)

は，$8000＋2n，$8000＋2n＋1番地を意味します。したがって

A:1)＝$AB

は，$8001番地の内容を$ABにすること，

A(1)＝$ABCD

は，$8003，$8002番地の内容を，それぞれ，$AB，$CDにすることを意味します。また，

B＝A:1)

は$8001番地の内容を変数Bの下位バイトの値とすること（上位バイトは0），

B＝A(1)

は，$8003，$8002番地の内容を変数Bの上位バイト，下位バイトの値とすることになります。

● 式

算術式は，定数や変数を以下の演算子で結合したものです。

＋（たし算）
－（ひき算）
＊（かけ算）
／（わり算）

そして，これらの間に優先順位はありませんから，式の評価は左から行なわれることになります。ただし，(　)を用いると，そこが優先されます。

(例)　1＋2＊3　の値は9
　　1＋(2＊3)　の値は7

また，論理式に使われる演算子は以下のようになっています。

＞　（大きい）
＞＝　（大きいか等しい）
＜　（小さい）
＜＝　（小さいか等しい）
＝　（等しい）
＜＞　（等しくない）

そして，論理式の値は真のとき－1，偽のとき0となります。

(例)　1＝1　の値は－1
　　1＝2　の値は0

その他の演算子としては，次のようなものがあります。

．（ビットごとのAND）
；（ビットごとのOR）
！（ビットごとのXOR）

(例)　〝A〟．$F0の値は$40，1！1の値は0

● 文

GAMEの文はほとんどが特殊変数への代入文となっています。すなわち，一般的な形式は次のようになります。

特殊変数＝式

具体的には表1.2に示すようなものがあります。

● 関数

GAMEの関数は表1.3に示すようなものがあります。関数の値は代入文などに使用します。

(例)　A＝$
　　R＝'10

● その他

表1.2　GAME-MZで使用できる文

文の種類	一般形	意味
代入文	変数＝式	式の値を変数に代入する。
IF	；＝式	式の値が0なら次の行へ移る。
GOTO	＃＝式	式で示す行番号へジャンプ。
NOP	＃＝0	何もしない。
STOP	＃＝－1	プログラムの実行を止める。
CALL	＞＝式	式で示される番地から始まる機械語サブルーチンのコール。
GOSUB	！＝式	式で示される行番号から始まるサブルーチンのコール。
RETURN	］	サブルーチンから戻る。
FOR …… NEXT	変数＝初期値，終値 …… @＝変数＋増分	初期値から始まる変数の値に増分を加えていき，終値を越えるまで間の文を繰り返す。
DO …… UNTIL	@ …… @＝(式)	式の値が0でなくなるまで，間の文を繰り返す。
PRINT	？　＝式 ？(桁)＝式 ？？＝式 ？$　＝式 $　＝式	式の値の10進左詰め出力。 式の値の10進右詰め出力(桁数を指定)。 式の値を16進4桁で出力。 式の値を16進2桁(下位バイト)で出力。 式の値の下位バイトをASCIIコードとみた文字の出力。
SPACE	．＝式	式の値と同じ数の空白を出力。
TAB	，＝式	式の値と同じ数だけカーソル位置を進める。
CRLF	／	改行する。
文字列出力	〝文字列〟	文字列を出力する。
SET RND	'＝式	乱数レジスタの初期値を式の値にする。
注釈		行番号のあと，空白をあけないとその行は注釈行。
マルチステートメント		文を空白で区切るとマルチステートメント。

表1.3　GAME-MZで使用できる関数

名前	一般形	引数・値
INPUT	？ $	キーボードから入力された式を値とする。 キーボードから入力された1文字のASCIIコードを値とする。
GET	［	押されたキーのASCIIコードを値とする。
RND	'式	1～式の間の乱数を値とする。

セーブ，ロード，ベリファイ以外にも，組み込みの機械語サブルーチンとして，ダウンスクロール，レフトスクロール，ライトスクロールがあります。それぞれのエントリー番地は，$5E40, $5E60, $5E70となっています。

(例)　＞＝$5E60

サンプルプログラム

GAMEは前にも述べましたようにゲームを作るのに適しています。そこでゲームを作ってみました。これは，昔ゼームセンターにあったボールをラケットで打ち返すという単純なゲームをもっと単純にしたものです。そのプログラムをリスト1に示します。その内容は，Zと Qのキーで左のラケット，=と?で右のラケットを操作して，どちらかが失敗するまでボールを交互に打ち返すというものです。

なお，プログラムの中で，1～4行は画面の色の設定，10～30行がリアルタイムキー入力のための機械語サブルーチンの作成を行なっています。

また，このプログラムではBASICらしさを出すために，マルチステートメントを多用したり，算術式と論理式の演算によるIF文の排除といったことをしています。このため，お世辞にも「読みやすい」プログラムとは言えないものになってしまいました。

リスト1　GAME-MZサンプルプログラム

```
 1 I=$D800,$DBE7 I:0)=6 @=I+1
 2 I=$D800,$DBE7 I:0)=39 I:39)=23 @=I+40
 3 I=$D801,$D826 I:0)=71 I:960)=71 @=I+1
 4 I=$D83B,$DBAB I:0)=7 @=I+40
10 A=$5EA0 A(0)=$0118 A(1)=$3AF0
20 A(2)=$5EA2 A(3)=$0032 A(4)=$3AE0
30 A(5)=$E001 A(6)=$A232 A(7)=$C95E
40 C=2*(´2) F=0 Q=F D=12 O=D
50 X=´21+8 Y=1 E=0 A=$5EA2 V=$D000
60 "■" ?=185 L=67 R=D S=0 ?=181
70 R=O S=39 ?=181
80 ?=300 ;=E<>0 H=1,5 >=$3E @=H+1 #=110
90 ?=800 ?=805 ?=805 ?=800
100 H=1,1000 @=H+1 #=80
110 ;=C<3 F=F+10 C=4 #=140
120 ;=C>2 Q=Q+10 C=2
140 "■" H=1,13 "■■" @=H+1 ?=F "    VS
 " ?=Q H=1,10000 @=H+1 ノ
160 "■■■■■■■■■PUSH ANY KEY FOR NEXT BALL"
170 H=$ H=1,5000 @=H+1 >=$3E #=50
180 M=Y N=X ?=200 ]
181 M=R N=S ?=200 .
182 M=R-1 ?=200 M=R+1 ?=200 ]
185 L=71 ?=180 .]
200 U:40*M+N)=L ]
300 ?=400 X=2*(C<3)+X-1 Y=(C/2*2=C)*2+Y-
1 ?=185 ?=500 ]
400 L=67*(E=0)*((X=0)+(X=39)) ?=180 ]
500 K=(12-Y)/12+C ;=C<>K C=K >=$3E
502 ;=X=0 E=(Y-D)/2 ;=E=0 C=C-2 >=$3E
503 ;=X=39 E=(Y-O)/2 ;=E=0 C=C+2 >=$3E
504 ]
700 ?=710 G=3-I-I ;=I=0 H=1,100 @=H+1 ]
701 M=R-G N=S L=0 ?=200
702 R=(I=1)*(R<23)-((I=2)*(R>1))+R
703 M=R+G L=67 ?=200 ]
710 A:0)=T >=$5EA0
711 ;=A:0)=U I=1 ]
712 A:0)=W >=$5EA0
713 ;=A:0)=Z I=2 ]
714 I=0 ]
800 R=D I=B T=$F1 U=$BF W=$F2 Z=$7F
801 S=0 ?=700 D=R B=I ]
805 R=O I=J T=$F7 U=$FD W=$F0 Z=$DF
806 S=39 ?=700 O=R J=I ]
```

ちょっとひと言

GAMEはそのモデルをTiny BASICに置いているため，BASICそのものがもつ欠点を暴露している言語と言えます。作者の大西氏自身ASCII誌上の連載で，GAMEはAlgol系の言語に取って替わられるべきマイコン初期の時代の言語であることを認めています。そのことが，同じ大西氏の提案によるTL/1に結びついていくことになったのは確かだと思われます。しかし，BASICが，死滅するどころか成長を遂げ，しかもその頭文字の意味である『初心者のための』というレベルをはるかに超えて存在している現在において，GAMEは使う価値がないと言い切るのは言い過ぎのような気もします。やはり，その実行速度の速さとメモリ効率の良さは十分評価に値するものでしょう。

〈参考文献〉

◦大西博　『GAME』　ASCII　78年5～10月号
◦間宮信義　『GAME-MZインタプリタ』　ASCII　79年10月号　◦間宮信義　『GAME-MZコンパイラ』　ASCII，79年10月号

PALL

概　要

PALLは79年12月号のI/O誌上でハドソンソフトによって発表されました。これは，マイコン界に構造化プログラミング論争が及んでくることを見越したものに違いありません。構造化プログラミングについては色々なことが言われていますが，その目的は，理解しやすくデバッグや修正が楽にできるようなプログラムを作ることにあります。このような目的を果たすためにはBASICでは不十分な点が多すぎます。そこで人々の目は構造化に適したAlgol系言語であるPASCALに向けられることになるのです。しかし，PASCALは言語自体がかなり大きいため，少なくともディスクがなければ利用するのは困難です。そこで，機能の面では標準PASCALからは大幅に劣りますが，MZ-80Kの基本システム（20KRAM）でだけでも構造化プログラミングの雰囲気を味わってもらおうというのが，PALLの設計者の意図だと思われます。PALLとはPractice of ALgol Language（Algol言語の練習）の略語だそうですが，このことからも，PALLがPASCALのための学習，トレーニング用言語だということがわかります。このように，PALLはTiny PASCALとは呼ばれていませんが，その分だけPASCALにとらわれない自由な設計がされていると言えるでしょう。

PALLは，UCSD PASCALがそうであったように，エディタで作成したソースプログラムを，中間コードにひとまず落とし，それを仮想スタックマシンと呼ばれる一種のプロセッサで解釈し実行するという方式を採用しています。これらのことはすべてON MEMORYで行なわれるため，PALLはコンパイラと言ってもインタプリタという感が強く，プログラムの開発が容易になっています。そのため，実行速度の面では我慢をしなければなりませんが，これはしかたのないことでしょう。

エディタ・システム関係

PALLのエディタはソースプログラムを作成するほかにプログラムのコンパイルと実行を管理しています。このため，PALLのエディタはPALLモニタと呼ばれています。表2.1にPALLモニタのコマンドの一覧表を示します。また，PALLのメモリマップは図2.1のようになっています。

表2.1　PALLモニタのコマンド

コマンド	一般形	意　味
Insert	I	ソースプログラムの入力。ソースが存在するときはアペンド。
Break	B	ラインポインタの直前へ1行挿入。
Position	P＝行番号	ラインポインタを指定の行へ設定。
Delete	D	ラインポインタの指す行を削除。
List	L	ラインポインタの指す行からのプログラムのリスト。
Search	S, 文字列	指定する文字列のサーチ。
Write	W	ソースプログラムのカセットテープへのセーブ。
Yank	Y	ソースプログラムのカセットテープからのロード。
Run	RUN	ソースプログラムのコンパイルと実行。
Bye	!	モニタへジャンプ。
New	&	ソースプログラムをすべて抹消する。

ソースプログラムを作るためには，このPALLモニタからIコマンドを打ち込んで開始します。このとき，BASICとは違い，行番号は付けません。このコマンドから抜けるためには SHIFT + BREAK を押します。作成したプログラムのリストはLコマンドで見ることができます。このコマンドの実行後にはスクリーンエディットをすることが可能になっています。なお，このときPALLモニタはプログラムを編集するための行番号を付けてリストを示します。この行番号はソースプログラムに1行挿入したり，ソースプログラムから1行削除したりするときに使用されるものです。たとえば，n行目のひとつ前に1行挿入する場合は，

```
P＝n
B
```

としたあと，文を書けばよいのです。また，n行目を削除するときは，

```
P＝n
D
```

とします。

ソースプログラムを実行するには

```
RUN
```

と入力します。また，プログラムはソースプログラムのレベルでカセットテープに保存できます。このためのコマンドがWとYです。

文　法

● プログラムの構造

プログラムの構造は，図2.2のように，トップダウンに読み下せる構造になっています。このため，手続きや関数はそれを使用するよりも前に定義されていなければなりません。したがって，メインルーチンはプログラムのいちばん最後に記述されることになります。メインプログラムはBEGINで始まりEND.で終わる一個の複合文です。メインルーチンの終わりのENDの．は必ず付けなければなりません。

図2.1　PALLのメモリマップ

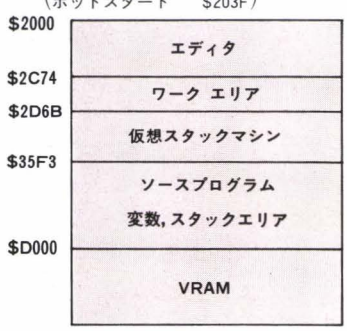
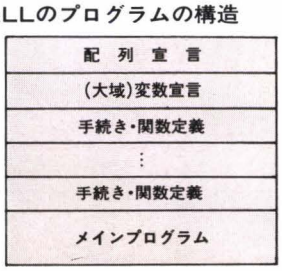


図2.2　PALLのプログラムの構造



● 定数・変数

PALLで扱うことのできるデータは10進数，16進数，1文字の3種類です。16進数定数，1文字は，10進定数と区別して，それぞれ，その前に$，〝を付けて表わします。

(例)　10，$1F，$12A0，〝A

変数名はA～Zの文字で始まる80字以内の文字や数字の列です。これは変数宣言部において

VAR　変数1，変数2，…，変数n；

の形で宣言しておきます。なお，プログラムの頭部で宣言される変数は，プログラム中のすべての場所で有効な大域変数，手続きや関数の中で宣言される変数は，その手続きや関数内だけで有効な局所変数です。大域変数と局所変数は名前が同じでも構いません。そのときは局所変数の値のほうが優先して使用されます。

● 配列

配列は，プログラム先頭の配列宣言部で，

ARRAY　配列名1[大きさ1]，…，配列名n[大きさn]；

の形で宣言することで使用することができます。ただし，これは整数型一次元配列のみです。たとえば，

ARRAY　A[10]；

とすれば，A[0]～A[9]でアクセスできる10ワード（1ワードは16ビット）の領域が確保されます。

ところで，配列宣言をしていなくても変数名に[　]を付けてメモリ上の領域をアクセスすることができます。たとえば，変数Bの値が$8000とすれば，B[n]というのは，メモリ上の$8000+2n，$8001+2n番地の領域を意味します。したがって，B[n]を代入文の左辺や右辺に使うことによって，POKE，PEEKの機能を実現することができます。

● 式

式とは，変数や定数を演算子で結び付けたものです。

PALLで使用できる算術演算子は次のようなものがあります。

＋　（たし算）
－　（ひき算）
＊　（かけ算）
／　（わり算の商）
MOD　（わり算の余り）

一方，論理演算子には次のようなものがあります。

＜　（大きい）
＞　（小さい）
＝　（等しい）
NE　（等しくない）

● 文

PALLで使うことのできる文を表2.2に示します。

ここで，手続きの宣言と関数の定義の仕方を説明しておきましょう。

手続きの定義は，

```
PROCEDURE 手続き名（引数1，…，
                    引数n）；
VAR　変数1，…，変数m；
BEGIN　文1；…；文ℓ　END；
```

の形式で，また，関数の定義は，

```
FUNCTION　関数名（引数1，…，引数
                    n）；
VAR　変数1，…，変数m；
BEGIN　文1；…；文ℓ END；
```

の形式で行ないます。VARによる変数宣言は，局所変数を宣言するものですが，それが必要ない場合は不要です。また引数がな

表2.2　PALLで使用できる文

文の種類	一　般　形	意　　味
代入文	変数←式	式の値を変数に代入する。
複合文	BEGIN 文1;…;文n END.	THENやELSE，DOなどの次には原則的にはひとつの文しかこないが，ひとつ以上の文を書くとき用いる。
手続き宣言文	PROCEDURE 名前(引数並び);	手続きの宣言をするために用いる。
関数宣言文	FUNCTION　名前(引数並び);	関数を宣言するために用いる。
RETURN文	RETURN　式;	式の値を関数の値として返す。
IF文	IF 条件 THEN 文1;	条件が真なら文1を実行する。
	IF 条件 THEN 文1 ELSE 文2;	条件が真なら文1を，そうでなければ文2を実行する。
REPEAT文	REPEAT 文 UNTIL 条件;	条件が真になるまで文を繰り返す。
WHILE文	WHILE 条件 DO 文;	条件が真の間，文を繰り返す。
WRITE文	WRITE（式1，…，式n）;	式の値をASCIIコードとみなして出力する。 式の代わりに2つの'で囲まれた文字列をもってくると，文字列をそのまま出力する。

い場合でも（と）は省略できません。さらに，関数宣言における文は，少なくともひとつが，

RETURN　式　；

の形のものでなければなりません。この文がないと関数から値を返すことができません。

もちろんこれらの定義では，PASCALの特徴のひとつである再帰的な定義をすることもできます。

（例）
```
FUNCTION  KAIJO(N);
BEGIN
IF  N<2  THEN  RETURN  1
ELSE RETURN N*KAIJO(N-1)
END;
```

余談ですが，この例などのように関数や手続きの本体がひとつの文の場合は，BEGINとENDで囲む必要はありません。しかし，筆者の好みでBEGINとENDをわざわざ付けてあります。

● 組み込み関数

PALLで最初から定義されている組み込み関数を表2.3に示します。そして，これらは式の中で使用します。

PALLでは，入出力において必要最小限のものしか関数や手続きとして存在していません（GETとWRITEだけ）。このため，10進数の入出力や16進数の入出力を行なう関数や手続きは自分で定義する必要があります。これらを定義することはプログラムのよい練習問題になるでしょう。

表2.3　PALLで使用できる関数

名　前	一般形	引 数・値
INC	INC（変数）	変数の値を1だけ増して，その変数に代入し，値とする。
DEC	DEC(変数)	変数の値を1だけ減じて，その変数に代入し，値とする。
GET	GET( )	1文字が入力されるまで待ち，押されたキーのASCIIコードを値とする。エコーバックはされない。

リスト2　$\sqrt{2}$の値を求めるプログラム

```
1 ARRAY A[90],B[90],X[90],Y[90];
2 VAR LNG;
3 PROCEDURE MOVE(A,B);
4  VAR I;
5  BEGIN
6   I←0;
7   WHILE I<LNG DO
8    BEGIN
9     A[I]←B[I];I←I+1;
10    END;
11  END;
12 PROCEDURE ADDI(A,N);
13  VAR M,I,Q,R,CRY;
14  BEGIN
15   M←N;CRY←0;Q←M/10;R←M MOD 10;
16   I←0;M←Q;
17   WHILE I<LNG DO
18    BEGIN
19     A[I]←A[I]+R+CRY;
20     IF A[I]>9 THEN
21      BEGIN
22       A[I]←A[I]-10;CRY←1;
23      END
24     ELSE
25      CRY←0;
26     Q←M/10;R←M MOD 10;M←Q;I←I+1;
27    END;
28  END;
29 PROCEDURE CLEAR(A);
30  VAR I;
31  BEGIN
32   I←0;
33   WHILE I<LNG DO
34    BEGIN
35     A[I]←0;I←I+1;
36    END;
37  END;
38 PROCEDURE COMPL(A);
39  VAR I;
40  BEGIN
41   I←0;
42   WHILE I<LNG DO

43    BEGIN
44     A[I]←9-A[I];I←I+1;
45    END;
46  END;
47 PROCEDURE SHIFT(A);
48  VAR I,S,T;
49  BEGIN
50   I←1;S←A[0];A[0]←0;
51   WHILE I<LNG DO
52    BEGIN
53     T←A[I];A[I]←S;S←T;I←I+1;
54    END;
55  END;
56 PROCEDURE ADD(A,B);
57  VAR CRY,I;
58  BEGIN
59   CRY←0;I←0;
60   WHILE I<LNG DO
61    BEGIN
62     A[I]←A[I]+B[I]+CRY;
63     IF A[I]>9 THEN
64      BEGIN
65       CRY←1;A[I]←A[I]-10;
66      END
67     ELSE
68      CRY←0;
69     I←I+1;
70    END;
71  END;
72 PROCEDURE MULTI(A,N);
73  VAR CRY,I;
74  BEGIN
75   CRY←0;I←0;
76   WHILE I<LNG DO
77    BEGIN
78     A[I]←A[I]*N+CRY;
79     IF A[I]>9 THEN
80      BEGIN
81       CRY←A[I]/10;
82       A[I]←A[I]-CRY*10;
83      END
84     ELSE CRY←0;
85     I←I+1;
86    END;
87  END;
88 PROCEDURE SUB(A,B);
89  BEGIN
90   COMPL(B);ADDI(B,1);ADD(A,B);
91   COMPL(B);ADDI(B,1);
92  END;
93 FUNCTION GETN();
94  VAR N;
95  BEGIN
96   N←1;MOVE(A,X);
97   MOVE(B,Y);SHIFT(B);MOVE(Y,B);
98   WHILE A[LNG-1] NE 9 DO
99    BEGIN
100     MOVE(A,X);MOVE(B,Y);ADDI(B,N);
101     MULTI(B,N); SUB(A,B);N←N+1;
102    END;
103  N←N-2;
104  ADDI(Y,N);MOVE(B,Y);MULTI(B,N);
105  SUB(X,B);SHIFT(X);SHIFT(X);
106  ADDI(Y,N);
107  RETURN N;
108 END;

109 PROCEDURE ROOT(M);
110  VAR N;
111  BEGIN
112   IF M>0 THEN
113    BEGIN
114     N←GETN();WRITE(N+48);
115     ROOT(M-1);
116    END
117   ELSE
118    WRITE($0D);
119  END;
120 BEGIN
121  LNG←60;CLEAR(X);CLEAR(Y);
122  ADDI(X,2);ROOT(50);
123 END.
```

サンプルプログラム

PALLを使って整数の平方根を求めるプログラムを作ってみました。方法は開平計算によっています。開平計算とは，以下のようなアルゴリズムです。

{ai} を整数Xの平方根の小数点以下 i 桁目の値とします（a_0は$a_0^2 \leqq X$であるような最大の整数です）。たとえば，X＝2なら {ai} は次のようになります。

$a_0=1$，$a_1=4$，$a_2=1$，$a_3=4$，$a_4=2$，…

のようなai（$i=_0,_2,\cdots$）は次のようにして組織的に求めることができるのです。

①　Y←0；i←0

②　$(Y+n)\times n \leqq X < (Y+n+1)\times(n+1)$ となるような整数nに対して，

ai←n；X←$(X-n^2)\times 100$；Y←$(Y+2n)\times 10$

③　i←i+1；goto　②

このとき，XやYを普通の変数とすると，すぐ桁あふれを起こしてしまいます（aiがひとつ求まるごとにXは2桁，Y は1桁増加しますから）。このため，X，Yは配列の形で表現し，多倍長の計算をすることにします。このようにして$\sqrt{2}$の値を50桁で求めるプログラムがリスト2です。リストの中のメインプログラム（121～123行）で，ADDI(X,2)をADDI(X,3)，ADDI(X,5)，…と変えると$\sqrt{3}$，$\sqrt{5}$，…の値を求めることもできます。また，求める桁数はROOTの引数を変えることで指定できます。このとき，LNGの値はその桁数より大きくしておく必要があります。

ちょっとひと言

このPALLは，メーカー側がPASCALの学習用と開き直っているところからもわかるように，始めから備わっている機能はあまり豊富ではありません。しかし，逆にそのシンプルさが魅力と言えるのではないでしょうか。命令の少なさは覚えるのにうってつけです。しかも，関数や手続きは必要なだけ，あとから定義できるのですから，使用していく上で不便はあまりないでしょう。

それにしても，当時の記事を見ると「実数も扱えるPALLIIを作成中」とありますが，いったいこれはどうなったのでしょうか。

〈参考文献〉

- 中本伸一　竹部隆司『PASCAL時代がやって来た！』I/O，79年12月号。
- 工藤裕司『PALLを解剖する！』I/O，80年1月～4月号。

FORM& FORTRAN-MZ

概　要

FORMは，PALLと同じくハドソンソフトによって開発された，整数型Tiny FORTRANです。これは80年5月号のI/O誌上で初めて発表されました。

FORTRANは，マイコンの言語としてはあまり使われていませんが，大型計算機用の言語としてはもっともポピュラーなものです。したがって，アプリケーションソフトも膨大な量が存在しています。大学でのプログラムの教育用の言語としては，決まってFORTRANです（現在ではPASCALに押され気味のような気もしますが）。このように有名な言語ですから，FORTRANのサブセットがマイコン用に開発されたのも当然なことでしょう。しかし，相手はマイコンですから，機能上かなりの制約を受けています。このため，このFORMはFORTRANというよりも，むしろBASICに近いものとなっています。本格的なFORTRANを，という人にとってFORMは少しもの足りないかもしれませんが，FORTRANの気分を味わうことはできるのではないでしょうか。

FORMで特徴的なことは，PALLが中間コードのインタプリタであったのに対し，即時実行可能なZ80の機械語オブジェクトを生成するという点でしょう。このように，FORMは「本当の」コンパイラなのです。

そして，'81年1月号のI／O誌上では，秋山実氏によって，このFORM に実数演算機能を付け加えた，FORTRAN-MZが発表されました。このFORTRAN-MZはFORMを拡張したものですから，FORMのソースプログラムは，変数名をすべて整数型に変えるだけで，FORTRAN-MZ上で走らせることができます。

余談ですが，FORMの名称は，FORTRANの名前がFOR mula TRAN slator（数式翻訳）の略であることにちなんで付けられたものと思われます。

エディタ・システム関係

先に述べましたように，FORMと FORTRAN-MZはほとんど同じものです。それで，以後では，特に区別しないときは，FORMという単語はFORTRAN-MZのことも意味していると思ってください。

PALLではPALLモニタがプログラムのエディットとコンパイルの両方を行なっていました。しかし，FORMはエディットするためのモードとコンパイルするためのモードに分かれています。FORM起動時は，コンパイル・モードになっています。まず，FORMのメモリマップを図3.1に示しておきましょう。

● コンパイル・モード

コンパイル・モードで使われるコマンドを表3.1に示します。プログラムを実行するには，一般的には

```
COMPIL
RUN
```

とします。しかし，ソースプログラムが大きすぎるとメモリエラーが出るので，このときは，ソースプログラムを一旦テープにセーブしておいてから，EXECコマンドで1ブロック(256バイト)ずつコンパイルしていきます。そしてRUNコマンドを実行します。

コンパイルされた機械語オブジェクトは

```
BSAVE　ファイル名
```

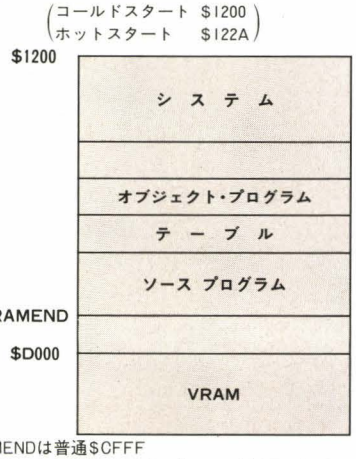
でカセットテープにセーブできます。これはモニタからロードすると直接実行することができます。ただし，FORTRAN-MZ では，システムがメモリ上に存在していなければ暴走してしまいます。

● エディット・モード

エディタコマンドの一覧表を表3.2に示します。このコマンドの機能は，PALLと似かよっていますが，Pコマンドはなく，コマンドのあとに行番号（エディタが付ける）を直接書くようになっています。また，このエディタもスクリーンエディットをすることができるようになっています。

プログラムを書くとき，実際のFORTR

図3.1　FORMのメモリマップ



RAMENDは普通$CFFF
（$27D8，$27D9番地を書き換えれば変更できる）

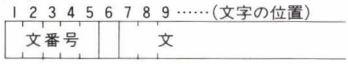
表3.1　コンパイラ・コマンド

コマンド	一般形	意　味
EDIT	EDIT	エディタ・モードへ移る。
COMPIL	COMPIL	メモリ上のソースプログラムをコンパイルする。
EXEC	EXECファイル名	カセットテープ上のソースプログラムをコンパイルする。ファイル名は省略してよい。
BSAVE	BSAVEファイル名	コンパイラが生成する機械語オブジェクトをカセットテープにセーブする。ファイル名は省略してよい。
LIST	LIST	1行コンパイルするごとにその行を表示させる。
	LISTN	コンパイルリストを表示しない。
	LISTE	エラーのある行だけを表示する。
	LISTP	すべての出力をプリンタへ切り換える。(FORTRAN-MZで，実数を出力するとき，それはプリンタへ出力されない)
	LISTC	プリンタ・モードを解除する。
RUN	RUN	オブジェクトプログラムの実行。
BYE	BYE	モニタへ制御を移す。

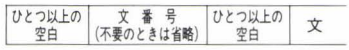
表3.2　エディタ・コマンド

コマンド	一般形	意　味
Insert	I	プログラムを入力できる状態にする。ソースプログラムが存在するときはその最後からインサートされる。
Break	B行番号	行番号で示される行の前へ1行インサートする。
List	L	プログラムをすべて表示する。
	L,行番号	行番号以降のプログラムを表示する。
	L,P	プリンタへプログラムを出力する。
Delete	D行番号	行番号で示される1行を削除する。
Compile	!	コンパイラ・モードへ移る。
New	&	ソースプログラムをすべて削除する。
Read	R ファイル名	カセットテープからソースプログラムをロードする。ファイル名は省略できる。
Write	W ファイル名	カセットテープにソースプログラムをセーブする。ファイル名は省略できる。

ANとFORMでは，その形式が大きく異なっています。つまり，FORTRANは1行の構成が，



でなければならないのに対し，FORMでは，



となっていて，自由な形式でプログラムを書くことができます。

文　法

● プログラムの構成

ソースプログラムは最初の行からコンパイルされます。したがって，原則的には，メインプログラムが最初に，その次にサブルーチンがこなければなりません。また，コンパイラはENDと書いてある行までをコンパイルしますから，FORMのプログラムは図3.2のような構成になります。

● 定数・変数

FORMが扱うことのできる数は10進数と$で始まる16進数です。また文字定数は1文字の前に＃を付けて表わします。

(例)　10，$8FFF，$0D，＃A

FORTRANでは，これに加えて，実数型の定数があります。これは，小数か，それにEで示される指数部を付けたものです。

(例)　1.23，−1.2，1.23E＋01(1.23×10^{1}のこと)，−2.3E−20(-2.3×10^{-20}のこと)

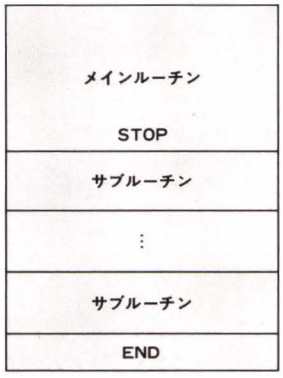
変数名は，1～4文字の英字で始まる英数字の列です。予約語（関数名やIF，WRITEなど，最初からシステムに定められているもの）は，変数名には使用できません。FORTRAN-MZでは，I, J, K, L, M, Nで始まる変数は整数型，他は実数型とみなします。

● 配列

配列は次のようなDIMENSION文によって宣言します。

DIMENSION　配列名1(大きさ)，…，
配列名n(大きさ)

図3.2　FORMのプログラムの構成



このとき，配列名は変数名と同じです。たとえば，

DIMENSION　A(10)

と宣言すれば，A(1)～A(10)で示される10個の領域が確保されます。なお，配列は2次元のものも宣言できます。このときは，

DIMENSION　A(10，20)

のようにします。

● 式

FORMでは論理式はありません（もちろん，実際のFORTRANではしっかりと存在しています)。したがって，式は算術式だけで，それらに使われる演算子には次のようなものがあります。

＋（たし算）
−（ひき算）
＊（かけ算）
／（わり算）

これに加えて，FORTRAN-MZには次のものがあります。

↑（べき乗）

演算子の優先順位は，

$$\uparrow \;>\; \begin{matrix} * \\ / \end{matrix} \;>\; \begin{matrix} + \\ - \end{matrix}$$

のようになっています。また，同じ優先順位のものは左から順に評価されていきます。

● 文

FORMの文の一覧表を表3.3に示します。この中で特色があるのはUSR文です。機械語サブルーチンをコールするときに，各レジスタの値をメモリ上のアドレスを介してセットできるようになっています。各レジス

表3.3　FORMの文(実行文)。非実行文は，DIMENSION文とEND行のみ。

文の種類	一　般　形	意　　味
代入分	変数＝式	式の値を変数や配列に代入する。
GOTO文	GOTO　文番号	指定した文番号(ユーザーが付けたもの)へジャンプする。
IF文	IF(式)文番号1, 文番号2, 文番号3	式の値が負なら文番号1，ゼロなら文番号2，正なら文番号3へジャンプする。文番号を省略すると次の行へ行く。
DO文	DO　文番号　変数＝初期値, 終値	以下の文と，文番号で示される文の間を，変数の値を初期値から1ずつ増加していき，終値を越えるまで繰り返す。
	DO 文番号 変数＝初期値, 終値, 増分	変数の増分が1以外のときのDO文。
CONTINUE文	CONTINUE	何もしない。
CALL文	CALL　文番号	指定した文番号から書かれているサブルーチンをコールする。
RETRN文	RETURN	サブルーチンから戻る。
PAUSE文	PAUSE　数字	数字を出力して，プログラムを一時停止する。 数字を省略すると，0が設定される。
STOP文	STOP　数字	数字を出力して，プログラムの実行をやめる。
BREAK文	BREAK	BREAKキーが押されていると実行をやめる。
USR文	USR(式)	式で指定される番地の機械語サブルーチンをコールする。
機械語文	$ML—機械語1, …, 機械語n	プログラム中に直接機械語を書き込む。
SETG文	SETG(式1, 式2)	X, Y座標が式1, 式2で与えられる画面上の座標をセットする。
RESG文	RESG(式1, 式2)	画面上の指定座標をリセットする。
READ文	READ(要素1, …, 要素n)	要素で指定される変数に値を入力する。1変数分入力するごとにCRキーを押す。
	〈要素の形〉	
	変数, I(または変数,)	I0進数入力。
	変数, B	I6進数入力。
	変数, A1	1文字入力。
	変数, A2	2文字入力。
	"文字列"	文字列を出力する。
	／	改行する。
	変数, E(FORTRAN-MZのみ)	実数入力。
WRITE文	WRITE(要素1, …, 要素n)	要素で指定される式の値を出力する。
	〈要素の形〉	
	式	I0進左詰め出力。
	式, In	n桁I0進左詰め出力。
	式, Bm	m桁I6進出力。m＝2, 4
	式, X	式の値だけスペースを出力。
	式, Am	m字の文字を出力。m＝1, 2
	式, V	式の値で示されるY座標へカーソルを移動する。
	式, H	式の値で示されるX座標へカーソルを移動する。
	"文字列"	文字列の出力。
	／	改行する。
	式, E(FORTRAN-MZのみ)	式の値のE型式出力。

タのアドレスは次のようになっています。

AF→$1000
BC→$1002
DE→$1004
HL→$1006

また，DIMENSION文とEND行には文番号を付けることはできません。

マルチステートメントはありません。このため，1行につき，ひとつの文を書くことしかできません。マルチステートメントはインタプリタの実行速度を上げるための役割をするものですから，コンパイラ言語にとっては不要でしょう。むしろ見やすいプログラムを書くことができます。

● 関数

FORMやFORTRAN-MZの組み込み関数を表3.4に示します。このうちMEMとIOCは代入文の＝の右側に書くことができます。

MEM(式1)＝式2　(メモリへのアクセス)
IOC　(式1)＝式2　(I/Oポートへのアクセス)

はそれぞれ，式1で示される番地やI/Oポートへ，式2の値の下位8ビットを書き込む機能があります。

表3.4　FORM, FORTRAN-MZの組み込み関数

関数名	一般形(FORM)	一般形(FORTRAN-MZ)	引数・値
MEM	MEM(式)	MEM(式)	式で示される番地の内容を値とする。
IOC	IOC (式)	IOC(式)	式で示されるポートの内容を値とする。
LOW	LOW(式)	LOW(式)	式の下位8ビットの値を返す。
GET	GET	GET	キーが押されていれば，そのASCIIコード，押されていなければ，ゼロを返す。
MOD	MOD(式1, 式2)	MOD(式1, 式2)	式1の値を式2の値で割ったときの余りを値とする。
ABS	ABS(式)	IABS(式)	式の値の絶対値。
	——	ABS(式)	式の値の絶対値(実数型)。
RND	RND(式)	IRND(式)	0～(式－1)の値の乱数。
SIGN	SIGN(式1, 式2)	ISIGN(式1, 式2)	式2の符号が付いた式1を値とする。
SQRT	——	SQRT(式)	式の値の平方根(実数型)。
三角関数	——	SIN(式)	ラジアン単位の式に対する正弦(実数型)。
	——	COS(式)	ラジアン単位の式に対する余弦(実数型)。
	——	TAN(式)	ラジアン単位の式に対する正接(実数型)。
指数関数	——	EXP(式)	自然対数の底eのべき乗(実数型)。
自然対数	——	ALOG(式)	式の値の自然対数(実数型)。
逆正接	——	ATAN(式)	逆正接(実数型)。
実数変換	——	FLOAT(式)	式の値の実数化(整数→実数)。
整数変換	——	IFIX(式)	式の値の整数化(実数→整数)。
OR	——	IOR(式1, 式2)	式1と式2のビットごとのOR。
AND	——	IAND(式1, 式2)	式1と式2のビットごとのAND。
EXCLUSIVE OR	——	IXOR(式1, 式2)	式1と式2のビットごとの排他的OR。

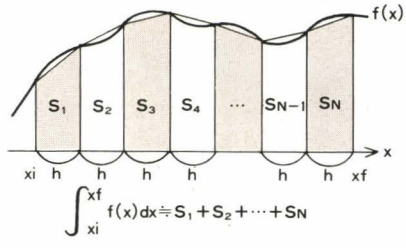
(注) 表の中の実数型というのは引数→値の変換が実数→実数のものを指す。何も書いてないものは整数→整数の変換。

サンプルプログラム

サンプルプログラムとして，FORTRAN-MZの実数演算機能を使った，数値積分を行なうプログラムを考えてみました。その原理は，台形公式と呼ばれているもっとも単純なものです。これは，関数f(x)が与えられたとき，その積分区間［x_i, x_f］をN等分して，各々の区間の面積を台形の面積で近似したものの総和を積分の値とするものです（図3.3参照）。

台形公式で積分の近似値を得たとき，誤差はEuler-Maclaurinの公式で知ることができます。すなわち，積分区間のきざみ幅を

図3.3　台形公式による積分の近似値



リスト3　Romberg積分を計算するプログラム

FORTRAN-MZによるもの。

```
      CALL 3000
      XI=0.0
      XF=1.0
      H=1.0
      P=(XF-XI)/H
      N=IFIX(P)
      WRITE(/"NO.     TRAPEZOIDAL
ROMBERG"//)
      CALL 1000
      S0=S
      DO 1 K=0,9
       WRITE(K,I1,2.X,S0.E)
       N=2*N
       H=0.5*H
       CALL 1000
       S1=S
       S=(4.0*S1-S0)/3.0
       WRITE(2.X,S.E/)
       S0=S1
     1 CONTINUE
       CALL 4000
       STOP
1000 S=0.0
       X0=XI
       DO 10 I=1,N
        X=X0
        CALL 2000
        F0=F
        XI=X0+H
        X=X1
        CALL 2000
        F1=F
        X0=X1
        S=S+0.5*(F0+F1)*H
    10 CONTINUE
       RETURN
2000 F=EXP(X)
       RETURN
3000 $ML F5,D5
       $ML 3E,00,11,00,00
       $ML CD,33,00
       $ML D1,F1
       RETURN
4000 $ML F5,D5
       $ML CD,3B,00
       $ML 7A
       CALL 4010
       $ML 7A
       CALL 4020
       $ML 7B
       CALL 4010
       $ML 7B
       CALL 4020
       $ML CD,06,00
       $ML D1,F1
       RETURN
4010 $ML 1F,1F,1F,1F
4020 $ML CD,DA,03,CD,12,00
       RETURN
       END
```

HuBASICによるもの。

```
1000 TIME$="00:00:00"
1010 XI=0
1020 XF=1
1030 H=1
1040 N=(XF-XI)/H
1050 LPRINT "NO.     TRAPEZOIDAL     ROMB
ERG";CHR$(13)
1060 GOSUB 2000
1070 S0=S
1080 FOR K=0 TO 9
1090  LPRINT K;TAB(5);S0;TAB(21);
1100  N=2*N
1110  H=.5*H
1120  GOSUB 2000
1130  S1=S
1140  S=(4*S1-S0)/3
1150  LPRINT S
1160  S0=S1
1170 NEXT K
1180 LPRINT TIME$
1190 STOP
2000 S=0
2010 X0=XI
2020 FOR I=1 TO N
2030  X=X0
2040  GOSUB 3000
2050  F0=F
2060  X1=X0+H
2070  X=X1
2080  GOSUB 3000
2090  F1=F
2100  X0=X1
2110  S=S+.5*(F0+F1)*H
2120 NEXT I
2130 RETURN
3000 F=EXP(X)
3010 RETURN
```

h ($h=\frac{x_f-x_i}{N}$) とすれば，誤差E(h)は，

$$E(h)=a_1h^2+a_2h^4+a_3h^6+\cdots$$

となります。したがって台形公式による値をI(h)とすれば，

$$I(h)=\int_{x_i}^{x_f}f(x)dx+a_1h^2+a_2h^4+\cdots$$

となります。このとき，

$$\int_{x_i}^{x_f}f(x)dx=\frac{1}{3}(4I(\frac{h}{2})-I(h))+\frac{3}{4}a_2h^4+\cdots$$

が成り立ちますから，I(h)，$I(\frac{h}{2})$ を求めて，$\frac{1}{3}(4I(\frac{h}{2})-I(h))$ を新しい積分値とすれば誤差を減らすことができます。このような積分のやり方はRomberg積分と呼ばれています。リスト3は，このRomberg積分を2000の文番号から始まる関数値を求めるサブルーチンに対して計算するプログラムです。なお，3000番以下，4000番以下はそれぞれ，内蔵タイマをセット，表示するサブルーチンです。また，FORTRAN-MZが，いかにBASICに近いかを知るため，それに対応するHu-BASICのプログラムも示してあります。両者のプログラムですべての計算が終わるまで（きざみ幅を半分ずつにしていって10回分Romberg積分をしている）の時間を比べると，FORTRAN-MZが5分32秒，Hu-BASICが3分です。この例では，FORTRAN-MZがいかにも遅いように思われますが，これはEXP(X)の計算の能率が悪いためと思われます。ためしに，積分する関数を，X，X＊X，X＊X＊Xで試すと，それぞれ，24秒，38秒，47秒と，Hu-BASICで同じことをやらせる場合の2～3倍のスピードでした。これから見てもFORTRAN-MZの高速さがわかります。コンパイラがインタプリタより速いのは当然であってほしいものですね。

ちょっとひと言

現在，JISで決められているFORTRANの標準はFORTRAN77ですが，FORMは，ひと昔前のJISの標準のうち，水準3000という最低レベルのものあたりをもとにしているようです。計算型のIF文などは生きた化石としか言えません。それに，メモリ上の問題から，SUBROUTINE文やFUNCTION文をなくし，引数による値の受け渡しを不可能にした点なども，言語の仕様だけの面から見れば，PALLから退化してしまったように思えます。現在においてFORMの価値を見出そうとするなら，高速なゲームを機械語よりも容易に開発できるということになるでしょう。

〈参考文献〉
・中本伸一　他　『FORM』I/O　80年5月～9月号。
・秋野実　『FORTRAN-MZ』　I/O　81年1月号。

microPASCAL-MZ

概　要

microPASCAL（以後μPASCALと書きます）は，丸山定信氏が8080用に開発したTiny PASCALです。そのソースリストは80年のASCII誌，1月号，2月号にアセンブリ言語の形で発表されました。これは，μPASCALの他機種への移植を積極的に促したもので，MZ-80には，辻田政広氏によって移植されました。それは，80年のASCII誌6月号で発表されています。

μPASCALは，PALLと同じく，ディスクを使わずに少量のメモリでPASCALを走らせることを目的としています。このため，標準PASCALと比べて，かなり機能が縮小されているという点はPALLと変わりありません。しかし，その機能はPALLよりは拡大されているといえます。たとえば，μPASCALでは，数値の入出力，プリンタへの出力ができるようになっています。それに付け加えて，機械語とのリンクが容易にできるような配慮（変数の番地を知ることができる，機械語サブルーチンを呼ぶとき，レジスタの値を簡単に設定できるなど）もされています。ただ，不平を言うとすれば，ソースプログラムのエディット時に，スクリーンエディットができないという点でしょう。

PASCALは，チューリッヒの連邦工科大学のN.Wirth教授によって考案されたプログラミングの教育用言語です。しかし，現在においては，単に教育用にとどまらず，マイクロプロセッサのシステム開発などの分野でも多く使われているみたいです。これは，プログラムの論理構造が明確であるため，読みやすく，理解しやすいプログラムを書くことができるからに違いありません。PASCALの需要はこれからもどんどん拡大していくことでしょう。

ところで，PASCALとは最初の計算機を発明したフランスの数学者B.Pascalにちなんで名付けられたものです。

エディタ・システム関係

μPASCALも他の言語と同じくシステム内にエディタを備えています。これは，ラインポインタによってソースプログラムを作成する完全なラインエディタです。そのため，スクリーンエディットに慣れている人にとっては少々扱いにくいものであるかもしれません。このμPASCALが好きになるかどうかは，このエディタを使いこなすことができるかどうかということにかかっているといっても過言ではないでしょう。表4.1にエディタとシステム関係のコマンドを示しておきます。また，μPASCAL-MZのメモリマップは図4.1のようになっています。

ここで，プログラムの作り方の手順を簡単に説明しておきます。一般的には，Bコマンドを入力することでプログラムを作り始めることができます。これはPALLやFORMのInsertコマンドに相当します。そして，

表4.1　μPASCALのエディタ・システムコマンド

命　令	一般形	意　味
Bottom	B	ポインタをプログラムの最後にもっていき挿入を続ける。
Delete	D行数	ポインタの指す行から，行数で指定されるだけの行を削除する。
Find	F文字列	指定された文字列で始まる行にポインタをセットする。
Insert	I	ポインタの指す行とその前の行の間に複数行を挿入する。
	I文字列	指定された文字列を，ポインタのひとつ前の行に挿入する。
Last	L行数	ポインタを，指定する行数だけ上の行に移す。
Next	N行数	ポインタを，指定する行数だけ下の行に移す。
Print	P行数	ポインタの指す行から指定した行数だけのリストを出して，ポインタを最後に印刷した行へ移す。
Hard copy	H行数	Pコマンドと同じだが，リストの出力がプリンタへになる。
Retype	R文字列	ポインタの指す行を指定する文字列で置き換える。
Top	T行数	ポインタをプログラムの先頭から，指定される行数だけ下の行へセットする。
Append	A	カセットテープから，ソースプログラムを読み込み，現在のプログラムの最後へアペンドする。
Save	S	ソースプログラムをカセットテープへセーブする。
Go	G／	ソースプログラムを実行する。
Kill	K／	ソースプログラムをすべて抹消する。
Quit	Q／	モニタへ制御を移す。
Memory	M	ファイル領域の残りバイト数を10進数で表示する。
End memory	Eアドレス	使用メモリのアドレスの終わり（より1だけ大きいアドレス）を16進数で指定する。

このエディットモードから抜けるときは SHIFT + BREAK によります。これで間違いなくプログラムを作ることができればよいのですが，入力ミスをしないですむことは稀にしかありません。そこで，Iコマンドと Rコマンドを使ってプログラムの修正をする段階となるわけです。

Iコマンドは，現在ラインポインタが指す行と，そのひとつ前の行の間に，プログラムを1行以上挿入するときに使います。複数行を一度に挿入できる点は，PALLや FORMのBreakコマンドと比べると，非常に便利なことと思われます。

Rコマンドは，ラインポインタが指している行を，Rの右に続く文字列と置き換えるものです。これは，1行のみを修正するのに使われます。そして，このときにこそ，スクリーンエディットができたら，と思わずにはいられないのですが，他人の作ったシステムに文句を言ってもしかたありません。しかし，このエディタには，プログラムのセーブ，ロード，プリンタへのリスト出力などひと通りの機能が備わっていますから，慣れてしまえば何の問題もなくなるでしょう。

文　法

● プログラムの構成

μPASCALのプログラムは，図4.2のようになっています。これは，PALLのプログラム構成とまったく同じになっています。

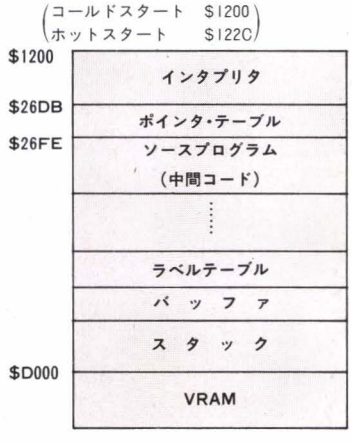
● 定数・変数

他のほとんどの小言語と同じく，μPASCALは整数型言語です。このため，使用される定数は，10進数，16進数，1文字です。そして，16進数は前に$を付けて，文字定数は1文字を〝または’で囲むことで表わすことができます。

(例)　1234，$FF，$12A0，〝A〟，‘B’

変数名は，英字で始まる英数字の列です。

図4.1　μPASCALのメモリマップ



変数を使うためには，あらかじめ

VAR　変数名1，…，変数名n；

とプログラムの頭部で宣言しておく必要があります。この変数は，すべての手続きや関数内で共通に利用できる大域変数になります。特定の手続きや関数の中でのみ有効な局所変数は，その手続きや関数の宣言と本体の間に，上記のようなVAR文を入れることで宣言することができます。

● 配列

配列の宣言は，プログラムの先頭で，

ARRAY　配列名1[大きさ1]，…，
　　配列名n[大きさn]；

によって宣言します。配列名は変数名と同じ規則で付けます。また配列は一次元のみしか許されていません。

● 手続き・関数の宣言

PALLにおいて，手続きや関数が，PROCEDURE文やFUNCTION文で定義されたように，μPASCALでは，それらをPROC文やFUNC文によって定義します。それらの形式は，

PROC　手続き名(引数1，…，引数ℓ)；
　VAR　局所変数1，…，局所変数m；
　　文；
FUNC　関数名(引数1，…，引数ℓ)；
　VAR　局所変数1，…，局所変数m；
　　文；

となります。PALLと定義の仕方が似ていますが，決定的な違いは，引数が存在しない場合は(…)の部分を省くということです。つまり，定義の最初が，

PROC　手続き名；

となります。こういうところは標準PASCALを真似てあるようです。また，局所変数が不要なときは，VAR文も必要ありません。文のところには，ひとつの文しか書くことができません。複数の文を書くときは，BEGINとENDで囲まれる複合文を使います。

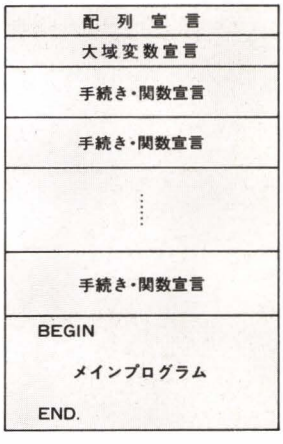
関数は値を返さねばなりません。PALLでは，これをRETURN文によっていましたが，μPASCALでは，FORTRAN(FORTRAN-MZではない）や標準PASCALと同じく，関数名に値を代入するという形で関数からの値を返します。このため，関数定義の本体には，

関数名：＝式；

という代入文が少なくともひとつ必要です。しかし，RETURN文とは違って，この代入文が実行されると直ちに関数からの値が返ってくるわけではなく，本体が終わったとき，最後に関数名に代入された値が関数の値となるので，このことは注意しておかなければなりません。

手続き，関数とも再帰的な定義ができることは，この種の言語では当然です。

図4.2　μPASCALのプログラム構成



● 式

式は定数や変数を演算子でつないだものです。これについて説明はいらないでしょう。μPASCALでは，以下のように多くの演算子を使うことができます。

◦算術演算子
- ＋　(たし算)
- －　(ひき算)
- ＊　(かけ算)
- ／　(わり算の商)
- DIV　(わり算の商
- MOD(わり算の余り)

◦比較演算子
- ＞＝　(大きいか等しい)
- ＞　(大きい)
- ＝　(等しい)
- ＃　(等しくない)
- ＜＝　(小さいか等しい)
- ＜　(小さい)

◦論理演算子
- OR　(論理和)
- AND (論理積)
- XOR (排他的論理和)

● 文

表4.2に，μPASCALで使用できる文の一覧表を示します。その中でも特徴的なのが，機械語サブルーチンを呼ぶためのCALL文でしょう。これは,呼ぶときにHLレジスタDEレジスタの値を代入文で設定できるようになっています。

ところで，PROCやFUNCによって定義された手続きは，

手続き名(式1，…，)；　|引数を持つもの|
手続き名；　|引数を持たないもの|
変数:＝関数名(式1，…)；|引数を持つもの|
変数:＝関数名；　|引数を持たないもの|

などとして使用します。

表4.2　μPASCALで使われる文

文の種類	一 般 形	意　　味
代入文	変数:＝式;	変数に式の値を代入する。
READ文	READ(変数,…);	数値をキーボードから変数に入力する。
WRITE文	WRITE(要素,…);	要素で示される文字列，数値を出力する。
	WRITE＃(要素,…);	WRITEの出力がプリンタへされる。
	〈要素の種類〉 式　：式の値を8桁右詰め出力。 式1:式2　：式1の値を式2の桁数で右詰め出力。 “文字列”　：文字列の出力。 ／　：改行。	
PUT文	PUT(要素,…);	要素が文字列のときはWRITEに同じ。 要素が式なら下位バイトをASCIIコードとみて出力する。
	PUT＃(要素,…);	PUTの出力がプリンタへされる。
複合文	BEGIN文1;…;文n END	複数の文をひとつの文とみなす。
IF 文	IF式 THEN文; IF式 THEN文1 ELSE文2;	式の値が真(0以外)なら文を実行する。 式の値が真(0以外)なら文1を，そうでなければ文2を実行する。
WHILE文	WHILE 式 DO 文;	式が真(0以外)の間だけ文を実行する。
REPEAT文	REPEAT 文1;…;文n UNTIL 式;	式が真(0以外)になるまでREPEATとUNTILの間を実行する。
FOR文	FOR変数:＝式1 TO 式2 DO 文;	変数の値を式1から1ずつ増していき，式2になるまで，文を繰り返す。
	FOR変数:＝式1 DOWN 式2 DO 文;	増分が－1であるようなFOR文。
注釈文	% 注釈 %	%で囲まれた部分は，読みとばされる。
STOP文	STOP;	プログラムの実行をやめる。
OUT文	OUT(式1):＝式2;	式1で示される出力ポートに，式2の下位バイトを出力する。
POKE文	POKE(式1):＝式2;	式1で示される番地に，式2の下位バイトを書き込む。
CALL文	CALL(式); CALL(式):＝式H; CALL(式):＝式H, 式D;	式で示される番地の機械語サブルーチンを呼ぶ。 HLレジスタに式Hの値を代入して機械語サブルーチンを呼ぶ。 HLレジスタに式Hの値，DEレジスタに式Dの値を代入して機械語サブルーチンを呼ぶ。

リスト4　データ整理のためのプログラム

```
ARRAY TBL[1000],BUF[81];
VAR TP;
PROC INIT;
 BEGIN
   TP:=0;
 END;
PROC MOVE;
 VAR I,A,C;
 BEGIN
   A:=ADRES(BUF[0])+1;
   I:=0;
   REPEAT
     C:=PEEK(A+I);
     TBL[40*TP+I]:=C;
     I:=I+1
   UNTIL C=$0D;
   TP:=TP+1
 END;
PROC GETL;
 BEGIN
   WRITE ("#");
   CALL(#03):=0,ADRES(BUF[0]);
   WRITE (/);
   MOVE
 END;
PROC PUTL (I);
 VAR C,J;
 BEGIN
   J:=40*I;
   REPEAT
     C:=TBL[J];PUT (C);PUT#(C);J:=J+1
   UNTIL C=$0D;
 END;
PROC OUTTBL;
 VAR I;
 BEGIN
   I:=0;
   WHILE I<TP DO
     BEGIN
       PUTL (I);I:=I+1
     END
 END;
PROC MAKTBL;
 BEGIN
   GETL;
   WHILE BUF[0]>*2320 DO
     GETL
 END;
FUNC COMP (I,J);
 BEGIN
   IF TBL[I]>TBL[J] THEN COMP:=1;
   IF TBL[I]<TBL[J] THEN COMP:=-1;
   IF TBL[I]=TBL[J]
     THEN
       IF (TBL[I]=$20) OR (TBL[I]=$0D)
       THEN COMP:=0
       ELSE
         COMP:=COMP (I+1,J+1);
 END;
PROC SWAP (I,J);
 VAR S,K,IK,JK;
 BEGIN
   K:=0;
   WHILE K<40 DO
     BEGIN
       IK:=I+K;JK:=J+K;

       S:=TBL[JK];TBL[JK]:=TBL[IK];
       TBL[IK]:=S;K:=K+1
     END
 END;
PROC SORT;
 VAR I,J,I1,I2;
 BEGIN
   FOR I:=(TP-3) DOWN 1 DO
     FOR J:=0 TO I DO
       BEGIN
         I1:=40*J;I2:=I1+40;
         IF COMP (I1,I2)>0 THEN
           SWAP (I1,I2)
       END
 END;
* MAIN PROGRAM *
BEGIN
  INIT;
  MAKTBL;
  SORT;
  OUTTBL
END.
```

● 組み込み関数

μPASCALの組み込み関数は表4.3のようなものがあります。

表4.3　μPASCALの組み込み関数

一 般 形	引 数・値
RND(式)	1～式の間の乱数を値とする。
ABS(式)	式の絶対値を値とする。
INP(式)	式で示される入力ポートから1バイト入力する。
GET	キーが押されていれば，そのキーのASCIIコード，押されてなければ0が値となる。
PEEK(式)	式で示される番地の内容を値とする。
ADRES(変数)	変数のアドレスを値とする。
INC(変数)	変数の値を1だけ増加させて，値とする。
DEC(変数)	変数の値を1だけ減少させて，値とする。

図4.3　入力データ（上側）を整理して出力（下側）したもの

(なお，このリストは，SORTの前に，一度OUTTBLを実行させることで打ち出した)。

```
LA-LA-LA              NAOKO K.   C-7
ABAYO                 NAOKO K.   C-22
ROUGE                 NAOMI C.   RK-51
USOTSUKI.GA.SUKIYO    MIMI H.    SV-6280
SHIAWASE.SHIBAI       JUNKO S.   SV-6319
OIKAKETE.YOKOHAMA     JUNKO S.   SV-6363
KONOSORA.O.TOBETARA   TOKIKO K.  DKQ-1028
KAMOME.WA.KAMOME      NAOKO K.   C-91
MADOGARASU            NAOKO K.   C-104
HATACHINI.NAREBA      JUNKO S.   SV-6468
MINIKUI.AHIRUNO.KO    NAOKO K.   C-119
AME...                RUMIKO K.  SM06-1
HITORIBOCCHIDE.ODORASETE NAOKO K. C-150
PIERROT               JINPACHI N. C-153
KESYOU                JUNKO S.   SV-7075
SUZUME                KEIKO M.   L.1567R
TABAKO                YOUKO K.   07SH1159

ABAYO                 NAOKO K.   C-22
AME...                RUMIKO K.  SM06-1
HATACHINI.NAREBA      JUNKO S.   SV-6468
HITORIBOCCHIDE.ODORASETE NAOKO K. C-150
KAMOME.WA.KAMOME      NAOKO K.   C-91
KESYOU                JUNKO S.   SV-7075
KONOSORA.O.TOBETARA   TOKIKO K.  DKQ-1028
LA-LA-LA              NAOKO K.   C-7
MADOGARASU            NAOKO K.   C-104
MINIKUI.AHIRUNO.KO    NAOKO K.   C-119
OIKAKETE.YOKOHAMA     JUNKO S.   SV-6363
PIERROT               JINPACHI N. C-153
ROUGE                 NAOMI C.   RK-51
SHIAWASE.SHIBAI       JUNKO S.   SV-6319

SUZUME                KEIKO M.   L.1567R
TABAKO                YOUKO K.   07SH1159
USOTSUKI.GA.SUKIYO    MIMI H.    SV-6280
```

サンプルプログラム

PASCALは，BASICと同じく汎用プログラミング言語と言えます。そこで，目的がサンプルであるようなプログラムは考えにくいのですが，ここでは，与えられたデータを整理するプログラムを作ってみました。といっても，大袈裟なものではなく，

項目名␣␣データ

の形で打ち込まれるデータを項目名で辞書式に並べ換えてプリンタへ出力するものです。このプログラムをリスト4に示します。MAKTBL が実行されると，データは1行(40字)ずつ，配列TBL内に読み込まれていきます。このとき，画面の左端には

#

の記号が出て，入力を待ちます。1カラム目に空白（以下のASCIIコードを持つ文字）が打ち込まれると，SORTに実行が移り，並べ換えが終わると，OUTTBLによってTBLの内容が画面とプリンタへ出力されます。それぞれの手続きのアルゴリズムは極めて単純ですから説明はいらないでしょう。なお，画面からの1行入力はモニタサブルーチン（$0003）を呼ぶことで行なっています。また，このプログラムの実行結果を図4.3に示しておきます。

ちょっとひと言

PALLにしろ，このμPASCALにしろ，PASCALの特色をなす，データの構造化といった機能を持っていません。また，ポインタというデータ型もありません。しかし，シャープが大々的に売り出している（いた？）インタプリタPASCALにもこういった機能は付いてなく，テープ版でPASCALを作ることの難しさが表われているように思われます。それならば，ということで機械語機能を充実させた，μPASCALは，マイコンを活用するには良い言語だと言えるのではないでしょうか。

〈参考文献〉

- 丸山定信 『micro PASCAL』ASCII, 80年1月号～3月号。
- 辻田政広 『micro PASCAL-MZ』ASCII, 80年6月号。

TL/1・MZ

（モニタ変換）

概　要

TL/1とはTiny Language 1（小言語No.1）の略です。これは，GAMEの製作者大西博氏によって製作されたAlgol系の小言語で，PASCALと似た文法を持っています。大西氏は，ASCIIでのGAMEの連載を終えるにあたって，GAMEは時代遅れで，これからはAlgol系の言語の時代であることを指摘していました。それに対する自分自身の答えというものが，このTL/1には表われているように思われます。すなわち，BASICには，局所変数が使えない，サブルーチンに引数が使えない，といった欠点があります。これでは中規模以上のプログラムを作るのは不便です。しかし，これらの機能をBASICに取り付けることと実行速度はトレードオフです。そこで，コンパイラを用いるわけになるのですが，それでは大量のメモリが必要になります。これでは実際の役には立ちません。そこで大西氏が考えたのは言語長を1バイトにして，ほとんどを1バイト命令で処理しようということです。思えば，GAME，PALL，FORM，microPASCAL，どれを取っても1語長は2バイトです。そのような状況にあって，ワードが1バイト長のコンパイラを作ってしまった大西氏の実行力には感服せずにはいられません。

TL/1が初めてASCII誌に掲載されたのは80年の6月号においてですが，それから約半年たった，81年の3月号で，MZへの移植版が発表されました。鈴木仁志氏の手によるものです。MZ-80K/CのモニタはSP-1002で，その機能はかなり貧弱でしたが，このTL/1・MZにはそれを補うための強力なモニタが内蔵されています。このため，このTL/1システムは一種のOS（言い過ぎ？）ともいえるほど，使い勝手が良くなっています。ただ，残念なことに1Z-009Aのモニタでは直接利用することができないことです。

エディタ・システム開発

TL/1において，コマンドは，モニタコマンドとエディタコマンドに分かれていますが，どのモードのときがどのコマンドといった区別はされていません。どちらのコマンドも同じコマンドレベルで使用することができます。この点が，モニタコマンドを使うにはモニタへ移る必要がある，Hu-BASICなどとは違っています。表5.1にモニタコマンドとエディタコマンドの一覧表を示します。

またメモリマップは図5.1のようになって

表5.1　TL/1のモニタコマンドとエディタコマンド

モニタコマンド

命　令	一般形	意味・機能
Memory Dump	Mn Mn, Mn, m	n番地から16バイト分のメモリの内容を表示する。 n番地から最後の番地までのメモリの内容を表示する。 n番地からm番地までのメモリの内容を表示する。 いずれの場合もSPACEで一時停止，SHIFT+BREAKで中断する。 このコマンドによる出力結果はスクリーンエディットができる。
Memory correct	:n a b…	n番地からのメモリの内容をa, b, …とする。
Go	Gn	n番地からの機械語サブルーチンを呼ぶ。
Decimal	Dn	nを10進数に変換して表示する。
Hexa decimal	Hn	nを16進数に変換して表示する。
In	In	n番の入力ポートの値を表示する。
Out	On, m	n番の出力ポートにmを書き込む。
Transfer	Tn, m, ℓ	n番地からm番地の内容をℓ番地からに移す。
Write	Wf, n, m Wf, n, m, ℓ	n番地からm番地までをfというファイルネームでテープにセーブする。 実行開始番地はn。 上と同じであるが，実行開始番地をℓにする。

エディタコマンド

命　令	一般形	意味・機能
List	0 n/	ソーステキストをすべて表示する。 行番号がn番からのソーステキストを表示する。 いずれも，一時停止，中断はMコマンドと同じ。
Delete	n	ソーステキストから行番号がnの行を削除する。
Insert	n文字列	文字列にnという行番号を付けて，テキストに入れる。
Auto	A An An, m	開始行番号1000，増分10で，プログラムを打ち込めるようにする。 開始行番号n，増分10で，プログラムを打ち込めるようにする。 開始行番号n，増分mで，プログラムを打ち込めるようにする。
Renumber	Rn, m	現在のプログラムを開始行番号n，増分mでリナンバする。 R, Rnの形もあるが，省略されているパラメータはAコマンドと同じ。
New	New	テキストをクリアする。
Question	Q	テキストの開始番地，最終番地，RAMENDの番地を表示する。
Begin pointer	B Bn	テキストの最終番地を正しくセットする。 テキストの開始番地をnとして，テキストの最終番地を正しくセットする。

表5.1 （つづき）エディタコマンド（つづき）

命　令	一般形	意味・機能
Find	F文字列	テキスト中にある，指定する文字列を含む行を表示する。 一時停止・中断はMコマンドと同じ。
Save	Sf	ソーステキストをfというファイルネームでテープにセーブする。 fは省略してもよい。
Load	Lf	テープからfというファイル名をロードし，現在のテキストにアペンドする。 fを省略すると，最初に見つかったファイルロードする。
Compile	C	ソーステキストをコンパイルする。
Execute	X	コンパイルされたオブジェクトプログラムを実行する。
Write	Wf	コンパイルされたオブジェクトをfというファイル名でテープにセーブする。 fは省略できる。

図5.1　TL/1・MZのメモリマップ

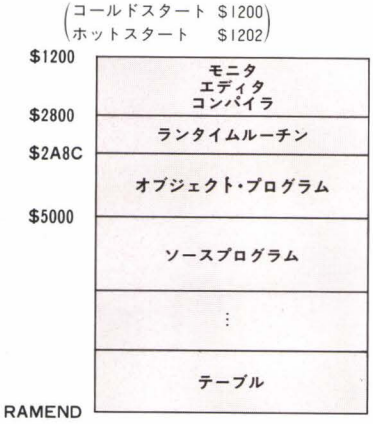
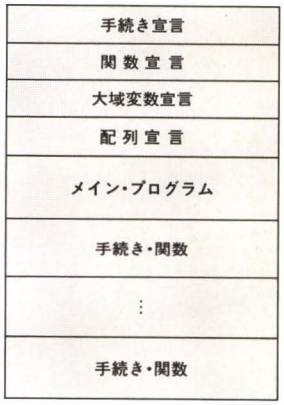


図5.2　TL/1のプログラム構成



います。

エディタは，行番号によってソースプログラムを管理するBASICタイプのものですから，BASICと同じ感覚でプログラムを打ち込むことができます。また，スクリーンエディットをすることもできます。

コマンドの一覧表を見れば，TL/1のモニタの素晴らしさがわかるのではないでしょうか。

文　法

● プログラムの構成

TL/1のプログラムは図5.2のような構成になっています。TL/1は，文法自体はPASCALに似ていますが，プログラムの構成はPASCALと大きく異なっています。まず，プログラムの先頭で，

PROC　手続き名1,…,手続き名n

によって，プログラム中に表われるすべての手続き名を宣言しておきます。同様に，次は，

FUNC　関数名1,…,　関数名m

によって，すべての関数名を宣言しておきます。もちろん，手続きや関数が存在しないときは省略します。

PASCALでは，ある手続きや関数がコンパイルされているとき，その中に表われる手続き名や関数名はすでに定義されていなければならないという鉄則があります。しかし，TL/1では，PROCやFUNC宣言を行なうために，手続きや関数の定義は，メインプログラムのあとなら，どういう順序で書いてもいいことになっています。

次にくるのが，大域変数宣言と配列宣言です。宣言の仕方は，VARとARRAYを使い，PALLやmicroPASCALとまったく同じですが，宣言の順序が逆ですから注意してください。

そして，その次にくるのがメインプログラムの本体です。PASCALなどでは，メインプログラムは，手続きや関数のあとにきますが，TL/1では最初にきます。プログラムを読むときは，普通，メインプログラムから読んでいきますから，TL/1のほうが自然な形をしているともいえます。メインプログラムはBEGINとENDによって囲まれた文の集まりですが，ENDの後に．は付いていません。

最後にくるのが手続きや関数の宣言です。手続きの宣言は次のような形をとります。

```
名前(引数1,…,引数n)
VAR　局所変数1,…,局所変数m
ARRAY　局所配列1〔サイズ1〕,…,
　　局所配列1〔サイズ1〕
BEGIN
　文1;文2;…
END
```

このとき，名前とは手続き名か関数名のことで，もはや，PROCやFUNCを付ける必要はありません。引数がない場合，(引数1,…）の部分は省略して名前だけを書きます。

局所変数の宣言や局所配列も，それらが不要な場合は省略できます。ところで，局所配列が宣言できるというのもTL/1の大きな特徴です。この宣言はPALLやmicroPASCALにはありません。

関数から値を返すためには，RETURN文で，

RETURN　式

という文を実行させます。これはPALLの場合と同じです。関数内では，RETURN文を少なくともひとつ書かねばなりません。

● 定数・変数

定数は1バイトの定数です。したがって10進数なら0～255(符号付きとみれば－128～127)，16進数なら，$00～$FF，文字定数なら‘1文字’で表わすことができるようになっています。論理定数は，真が$FF，偽が$00です。

● 演算子

TL/1の演算子は以下のようなものがあります。英字の演算子については，両側の．を省略しても構いません。

◦算術演算子

- ＋　（たし算）
- －　（ひき算）
- ＊　（かけ算，結果の上位バイトはMHIGHというシステム固有の変数に入る）
- ／　（わり算，余りはMODというシステム固有の変数に入る）
- .ADC.（キャリー付きのたし算）
- .SBC.（ボロー付きのひき算）

◦比較演算子

- ＞　（大きい）
- .GT.（大きい，数を符号付きとみるとき）
- ＜　（小さい）
- .LT.（小さい，数を符号付きとみるとき）
- ＝　（等しい）
- ＃　（等しくない）

◦論理演算子

- .AND.　（論理積）
- .OR.　（論理和）
- .EOR.　（排他的論理和）

● 文

TL/1で使用する（実行）文を表5.2にまとめました。

その他に注目すべきことは，複号文を表わすときは，BEGIN ENDの組みだけでなく，［　］，（　）の組が使えることです。これらを適当に使い分ければ，かなり見やすいプログラムを書くことができるでしょう。

表5.2 TL/1・MZの文

文の種類	一般形	意味
代入文	変数:=式	式の値を変数に代入する。
複合文	BEGIN 文1 文2 …END	複数の文をひとつの文とみなす。
IF文	IF 条件 THEN 文 IF 条件 THEN 文1 ELSE 文2	条件が真なら文を実行する。 条件が真なら文1, そうでなければ文2を実行する。
CASE文	CASE 式0 OF 式1 文1 ⋮ 式n−1 文n−1 ELSE 文n	式0の値が式i(1≦i≦n−1)の値と等しければ文iを実行する。そうでなければ文nを実行する。
FOR文	FOR 変数:=式1 TO 式2 DO 文 FOR 変数:=式1 DOWN 式2 DO 文	変数の値を式1の値から1ずつ増やし, 式2の値を越えるまで文を繰り返す。 上のFOR文で増分が−1のもの。
REPEAT文	REPEAT 文1 文2 …UNTIL条件	条件が真になるまでREPEATとUNTILの間を繰り返す。
WHILE文	WHILE 条件 DO 文	条件が真の間, 文を繰り返す。
POKE文	MEM(式1, 式2):=式	式1, 式2の値をそれぞれ上位, 下位アドレスとするメモリ領域に式の値を書き込む。
OUT文	PORT (式1):=式2	式1で示される出力ポートへ式2の値を出力する。
CALL文	CALL(数1, 数2) CALL(数1, 数2, 式1) CALL(数1, 数2, 式1, 式2) CALL(数1, 数2, 式1, 式2, 式3)	数1, 数2を上位, 下位バイトとする番地の機械語サブルーチンを呼ぶ。 式1の値をAレジスタに入れて機械語サブルーチンを呼ぶ。 式1の値をAレジスタ, 式2の値をHレジスタに入れてサブルーチンを呼ぶ。 式1, 式2, 式3の値をA, H, Lレジスタに入れてサブルーチンを呼ぶ。
SET文	SET (式1, 式2)	式1, 式2の値をx, y座標とする点のセット。
RESET文	RESET(式1, 式2)	式1, 式2の値をx, y座標とする点のリセット。
INVERSE文	INVERSE(式1, 式2)	式1, 式2の値をx, y座標とする点の状態を反転する。
SENSE文	SENSE	SHIFT+BREAK が押されていれば実行をやめる。
CURSOR文	CURSOR(式1, 式2)	式1, 式2で示される位置にカーソルを移動する。
STOP文	STOP	プログラムの実行をやめる。
RETURN文	RETURN RETURN 式	手続きから強制的に戻る。 式の値を関数の値として, 関数から戻る。
WRITE文	WRITE(式:要素1, …, 要素n)	要素を式で表わされる出力装置(0→CRT, 1→プリンタ)へ出力する。

〈要素の種類〉

式	式の値を10進左詰め出力。	SPACE(式)	式の値の数だけ空白を出力。
#(式1, 式2)	式2の値を式1の桁数で右詰め出力。	CRLF	改行する。
"文字列"	文字列の出力。	CRLF(式)	式の値の回数だけ改行する。
ASCII(式)	式の値をASCIIコードで出力。		

また, PASCALなどでは文の区切りを；で示していましたが, TL/1では空白を使ってもよいことになっています。したがって構文解析において, TL/1はPASCALに勝っているといえます。

● 組み込み関数

表5.3にTL/1で使える組み込み関数を示します。このうち, 特徴的なのはUSR関数でしょう。機械語サブルーチンを実行したときの結果（Aレジスタの値）を変数に代入できるのですから, 機械語とのリンクが非常にやりやすくなっています。

表5.3 TL/1・MZの組み込み関数

名前	引数・値
RND(式)	1～式の値の乱数を値とする。
NOT(式)	式の値の1の補数を値とする。
NEG(式)	式の値の2の補数を値とする。
COM(式)	式の値の1の補数を値とする。
ROR(式)	キャリー→式の最上位ビット, 式の最下位ビット→キャリーとなる右ローテート。
LSR(式)	0→式の最上位ビット, 式の最下位ビット→キャリーとなる右シフト。
ASR(式)	式の最上位ビットは変化しない, 式の最下位ビット→キャリーとなる右シフト。
ASL(式)	式の最下位ビット←0, キャリー←最上位ビットとなる左シフト。
ROL(式)	式の最下位ビット←キャリー, キャリー←最上位ビットとなる左ローテート。
RRC(式)	式の右ローテート。最下位ビット→最上位ビット。
RLC(式)	式の左ローテート。最下位ビット←最上位ビット。
USR(数1, 数2) USR(数1, 数2, 式1) USR(数1, 数2, 式1, 式2) USR(数1, 数2, 式1, 式2, 式3)	それぞれCALLと同じ機能を持つが, リターン時のAレジスタの内容が関数の値となる。
POINT(式1, 式2)	式1, 式2の値をx, y座標とする点がセットされていると真($FF)そうでなければ偽($00)を返す。
GET(0)	キーボードから1文字入力して, そのキーのASCIIコードを値とする。
READ(0)	キーボードから10進数を入力して, それを値とする。
RDHEX(0)	キーボードから2桁の16進数を入力して, それを値とする。
PORT(式)	式で示される入力ポートの値を返す。
MEM(式1, 式2)	式1, 式2の値を上位, 下位アドレスとするメモリの内容を値とする。

サンプルプログラム

TL/1のモニタは, いろんな機能をもっていますが, ソースプログラムのリストをプリンタへ出力することはできません。そこで, ソースリストをプリンタへ出力するためのプログラムを作りました。これは, あらかじめカセットテープにセーブされているソースプログラムを読み込んでプリンタへ出力するものです。そのリストはリスト5のようになっています。この中で機械語

リスト5　ソースプログラムをプリンタへ出力するプログラム

```
1000  PROC RDINF,RDDAT,PUTN,PUTL
1010  FUNC GETCH
1020  VAR EOL,EOF,TXTAH,TXTAL,C
1030  BEGIN
1040   TXTAH:=$50 TXTAL:=$00
1050   WRITE(0:"EOL CODE?")
1060    EOL:=RDHEX(0)
1070   WRITE(0:"EOF CODE?")
1080    EOF:=RDHEX(0)
1090   RDINF RDDAT
1100   C:=GETCH
1110   WHILE C # EOF DO
1120    [ PUTN PUTL
1130      C:=GETCH        ]
1140   WRITE(0:CRLF(2))
1150   WRITE(1:CRLF(2))
1160  END
1170  *
1180  RDINF
1190   VAR I,C
1200   BEGIN
1210    CALL($04,$D8)
1220    WRITE(0:CRLF,"LOADING ")
1230    WRITE(1:"LISTING OF ")
1240    I:=$F1 C:=MEM($10,I)
1250    WHILE C # $0D DO
1260     [ WRITE(0:ASCII(C))
1270       WRITE(1:ASCII(C))
1280       I:=I+1 C:=MEM($10,I)  ]
1290    WRITE(0:CRLF) WRITE(1:CRLF(2))
1300   END
1310  *
1320  RDDAT
1330   BEGIN
1340    MEM($11,$04):=$00
1350    MEM($11,$05):=$50
1360    CALL($04,$F8)
1370   END
1380  *
1390  GETCH
1400   VAR C
1410   BEGIN
1420    C:=MEM(TXTAH,TXTAL)
1430    TXTAL:=TXTAL+1
1440    TXTAH:=TXTAH.ADC.0
1450    RETURN C
1460   END
1470  *
1480  PUTN
1490   VAR I,D
1500   BEGIN
1510    D:=GETCH
1520    CALL($40,0,0,C,D)
1530    FOR I:=0 TO 3 DO
1540     [ D:=MEM($40,$40+I)
1550       WRITE(0:ASCII(D))
1560       WRITE(1:ASCII(D)) ]
1570    WRITE(0:SPACE(1))
1580    WRITE(1:SPACE(1))
1590   END
1600  *
1610  PUTL
1620   BEGIN
1630    C:=GETCH

1640    WHILE C # EOL DO
1650     [ WRITE(0:ASCII(C))
1660       WRITE(1:ASCII(C))
1670       C:=GETCH            ]
1680    WRITE(0:CRLF)
1690    WRITE(1:CRLF)
1700    SENSE
1710   END

$4000 F5 C5 D5 44 00 4D 00 E5:...D.M..
$4008 CD 2D 40 11 00 00 CD 4C:.-@....L
$4010 40 CB 78 20 03 13 18 F6:@.x .....
$4018 CD 45 40 79 F6 30 77 2B:.E@y.0w+
$4020 7A B3 28 04 42 4B 18 E3:z.(.BK..
$4028 E1 D1 C1 F1 C9 21 3E 40:.....!>@
$4030 3E 20 1E 06 77 23 1D 20:> ..w#. 
$4038 FB 3E 0D 77 2B C9 00 00:.>.w+....
$4040 00 00 00 00 0D 3E 0A 03:.....>..
$4048 3D 20 FC C9 3E 0A 0B 3D:= ..>..=
$4050 20 FC C9 00 00 00 00 00: .......
```

の部分は，2バイトの数値で表わされている行番号をASCIIコードの列に直すものです。この機械語プログラムは，TL/1のモニタを使えば簡単に打ち込むことができます。

プログラムをコンパイルし，実行させると，EOL（End Of Line）とEOF（End Of File）のコードを尋ねてきますから，ソースプログラムがTL/1のときは00とFFを順に入力してください。また，これと同じコードで，GAME-MZのソースプログラムをプリンタへ出力することも可能です。PALLやFORMのソースリストを出力するときは，EOLとEOFを0Dと00にすればよいでしょう。

なお，このプログラム中で呼んでいる機械語サブルーチンのうち，$04D8はカセットテープからインフォメーションブロックを読み取るルーチン，$04F8はデータブロックを，$10F1番地から始まるテーブルに従ってロードするルーチンです。

ちょっとひと言

TL/1は，それ以前のPALLやmicro PASCALなどの小言語を集大成したような言語であると言えます。もちろん，現在の恐ろしいまでに豊富な機能をもったBASICにはかなわない部分もあるかもしれませんが，当時としては，整数型小言語の行き着く先がこのTL/1だったのではないでしょうか。その証拠に，TL/1以来，それと同系統の言語でこれといったものは発表されてはいません。今でも，このTL/1を，モニタ変換することなくモニタ1Z-009Aの上で動くように移植することができれば相当な価値があると思うのですが，MZ誌上でそれを行なうのは版権の問題からみても不可能のように思えます。

〈参考文献〉

- 大西博他　『TL/1』　ASCII，80年6月号～11月号。
- 鈴木仁志他　『TL/1・MZ&TL/1・CBM』　ASCII，81年3月号。
- TL/1リファレンス・マニュアル　ASCII，82年10月号。

LISP

（I/O版はモニタ変換）

概　要

最近，人工知能の研究が盛んになってきていますが，それに伴って，それを助ける言語としてLISPやPROLOGが注目を浴びています。LISPはFORTRANに続いて古い言語ですが，やっとその価値が一般に認められてきたといえます。

LISPの大きな特徴は代入文がないということです。これはBASICなどの言語に慣れてしまっている人々にとっては戸惑いの原因になります。このため，LISPは，学ぶのが簡単であるにもかかわらず，敬遠されがちなのでしょう。また，LISPには関数だけしか存在しません。LISPにおいて，プログラムを作ることは，すでにある関数を寄せ集めて新しい関数を定義することにほかならないのです。したがって，BASICをよく知っている人で，新たにLISPを学ぼうとしている人は，これからPASCALを学ぼうとしている人よりも大幅な意識の改革が必要になります。しかし，LISPは記号処理言語の中ではもっとも有名なものですし，そこに流れている思想は他の記号処理言語にも相通じるものがありますから，知っていても損をしない言語です。そうするうちに，LISPの構造の数学的美しさ（単純さ）が理解できるようになるでしょう。

さて，MZで使えるLISPとしては，80年12月号のI/O誌上で，浅見俊幸氏らによって発表されたものが有名です。それは基本5関数だけからなる純LISPに数値計算とプリンタ機能を付け加えた簡単なものでしたが，かなり複雑なことをやらせることができ，LISPの面目を保っていました。しかし，このLISPは使い勝手の点からみると，あまりよいものとは言えません。プログラムを入力するときが，大変面倒です（修正がやりにくい！）。LISPをMZで使いたいと考えていた人々は，もっと使いやすいLISPの出現を待ち望んでいたに違いありません。そんなとき，Oh！PC83年4月号において，小倉哲氏によってPC88版のLISPが発表されました。そのLISPはI/Oのものと比べると格段の機能をもっていました。それを見た岡本明弘氏が，88版LISPをMZ用に移植しました。これは，MZ-700の単行本で発表される予定になっています。以下の節からは，これらふたつのLISPについて紹介をしたいと思います。

LISPの扱うデータ

LISPとはLISt Processor（リスト処理言語）の省略形です。そして，その名前のとおり，LISPが処理するデータはリストしかありません。しかも，それはリストの中でももっとも単純な構造をもつ二進木リストと呼ばれるものです。これは，図6.1のように，ひとつの節点からふたつの枝が出ているような構造をしています。そして，LISPはこの二進木リストを表現するのにS式（S expression）というものを使用します。これが，LISPにおいてよく見かけるカッコの集まりなのです。図6.1の二進木リストをS式で表現すると，

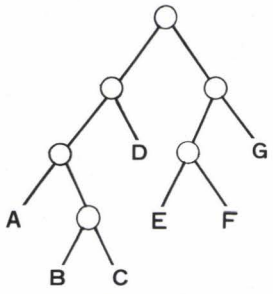
(((A␣(B␣C))␣D)((E␣F)␣G))

となります。これは，一見複雑に見えますが次のように考えると簡単です。

① は，(X␣Y) と表わす。

② XまたはYがさらに という形を

図6.1　二進木リスト



していたら，それを(U␣V)で置き換える。このように，二進木リストは再帰的にS式へ書き直すことができます。逆のことも同様にできますが，同じS式をもつといっても，それらが同じ二進木リストだとは限りません。S式から二進木リストの形は一意に決まりますが，まったく同じ形をもった二進木リストがメモリのふたつ以上の場所で同時に存在している可能性があるからです。

リストの末端にある単語は，アトムと呼ばれるS式です。アトムも厳密にはリストの形をしているのですが，さしあたっては，リストの要素のうちで，それ以上分解できないものと思っていればいいでしょう。

LISPの組み込み関数

LISPにはS式しかありません。関数が扱うデータも関数自身もS式なのです。したがって，データも関数も区別がありません。LISPは入力されたデータをすべて関数とみなして，その値（これもS式）を求めようとします。このため，S式を関数の引数として与えるときは，これは，データだから値を求めようとしてはいけない，あるいは，このS式の値はS式自身だ，ということを教えてやる必要があります。このための関数が，QUOTEなのです。

LISPの関数は，組み込み関数であろうとあとから定義したものであろうと，

(関数名␣S1␣S2␣…)

という形で表わします。S1, S2, …は引数として与えるS式です。

さて，QUOTEに話を戻します。いま，

(PLUS␣1␣2␣3)

というS式があるとします。PLUSとは引数の総和を値とする関数ですから，上のS式の値は6（数もそれ自身を値とするS式です）になります。しかし，

(QUOTE(PLUS␣1␣2␣3))

の値は，

(PLUS␣1␣2␣3)

というS式になります。

QUOTEの逆の働きをする関数がEVALです。EVALは引数として与えられたS式

表6.1　LISPの組み込み関数　(注)x, y, x_1, …x_nは二進木リスト，またはその式のS式。n, m, n_1, …, n_mは数値アトム。

〈I/O版と新版で名前が同じもの〉

名　前	引　数	値・機　能
CAR	x	xの第1要素。
CDR	x	xから第1要素を取り除いたもの。
NULL	x	xがNILならT，そうでなければNIL。
EQ	x, y	xとyが同じ二進木リストならT，そうでなければNIL。
EQUAL	x, y	xとyが同じS式ならT，そうでなければNIL。
ATOM	x	xがアトムならT，そうでなければNIL。
CONS	x, y	xを左，yを右の枝とするような新しい二進木リストを作り値とする。
QUOTE	x	xそのもの。
EVAL	x	xに対応するS式の値を求める。
PRINT	x	xをS式になおして出力する。また，それを値とする。
GET	x, y	xの中にyをさがし，あればその次の要素を値とする。なければNIL。
COND	$(x_1, y_1)\cdots(x_n, y_n)$	x_1がNILでなければy_1，x_2がNILでなければy_2，…を実行する。
PLUS	$n_1, n_2, \cdots, n_m$	$n_1+n_2+\cdots+n_m$
DIFFERENCE	n, m	n−m
ADD1	m	m+1
SUB1	m	m−1
LIST	$x_1, x_2, \cdots, x_n$	$(x_1, x_2, \cdots, x_n)$
MINUS	m	−m
MINUSP	m	m<0ならT，そうでなければNIL。
ZEROP	m	m=0ならT，そうでなければNIL。
GREATERP	m, n	m>nならT，そうでなければNIL。
LESSP	m, n	m<nならT，そうでなければNIL。
TIMES	$n_1, n_2, \cdots, x_m$	$n_1\times n_2\times\cdots\times n_m$。
APPEND	x, y	xとyをくっつける。
MEMBER	x, y	yの中にxという要素があればT，そうでなければNIL。

〈I/O版のみ〉

名　前	引　数	値・機　能
ASSOC	x, y	対リストyの中でxの対をみつけて値とする。なければNIL。
PAIRLIS	x, y, a	x, yからできる対リストと，対リストaをAPPENDする。
SUBLIS	a, x	xの中のアトムを対リストaの中の対応するものに置き換える。
NUMBER	x	xが数値アトムならT，そうでなければNIL。
QUOTIENT	n, m	nをmで割った商を値とする。
REMAINDER	n, m	nをmで割った余り。
EQN	n, m	n=mならT，そうでなければNIL。

〈新版のみ〉

名　前	引　数	値・機　能
CAAR	x	(CAR(CAR x))
CADR	x	(CAR(CDR x))
CDAR	x	(CDR(CAR x))
CDDR	x	(CDR(CDR x))
READ	ナシ	S式を読み込んで，二進木リストに変換して値とする。
ERROR	ナシ	この関数が呼ばれた場所(番地)を出力してトップレベルへもどす。
PAIR	x, y	xとyから対リストを作る。
RPLACA	x, y	xの左の枝をyで置き換える。
RPLACD	x, y	xの右の枝をyで置き換える。
SASSOC	x, y	I/O版のASSOCと同じ。
PAIRASS	x, y, a	I/O版のPAIRLISと同じ。
ONEP	n	n=1ならT，そうでなければNIL。
TERPRI	ナシ	改行する。
PUT	x, y, z	xの中にyを探し，その次の要素をzとする。
PHEX	n	nを16進数で出力する。
ADR	x	アトムxの番地を値とする。
ASC	n	nというASCIIコードをもつ文字を値とする。
PRIN1	x	アトムxを出力するが，改行しない。
LENGTH	x	リストxの要素の個数を値とする。
AND	$x_1, x_2, \cdots, x_n$	n=0ならNIL。xi(i=1〜n)のうちNILがあればNIL。そうでなければT。
OR	$x_1, x_2, \cdots, x_n$	n=0ならNIL。xi(i=1〜n)のうちTがあればT。そうでなければNIL。

表6.2　システムコントロール用の関数の使い方

I/O版	新　版	意　味
(OBLIST)	OBLIST*)	現在までに入力されているすべてのアトムのリスト。
(TRACE)	(TRACE T)	以後,関数が実行される毎にトレースする。
(CLEART)	(TRACE NIL)	関数のトレースをやめる。
(PRINTER)	(LP T)	出力をプリンタへ出す**)
(CLEARP)	(LP NIL)	プリンタへの出力をやめる。
(CLEARC)	——	今まで定義した関数を抹消する。
——	(RECLAIM)	無理やり,ガーベジコレクションを行わせる。
——	(GCP T)	ガーベジコレクション時のメッセージを出力する。
——	(GCP NIL)	ガーベジコレクション時のメッセージをやめる。
——	(END)	モニタへ制御を移す。

(注)　*)　新版LISPではOBLISTは関数ではなく定数。このため(　)は付けない。
　　**) I/O版が画面全てを出すのに対し,新版は出力結果のみ。
　　関数の定義は『関数の定義』の節へゆずる。

の値を強引に求めようとします。したがって

```
(EVAL␣(QUOTE␣(PLUS␣1␣2
␣3)))
```

の値は6となります。

表6.1にLISPの組み込み関数を示しておきます。なお,I/O版LISPは関数でないS式を引数として与えるときは,自動的にQUOTEが付いているとみなすので,QUOTEは必要ありませんが,統一性を欠いているように思われます。

(例)

```
(CAR␣(A␣B␣C))          (I/O版)
(CAR␣(QUOTE␣(A␣B␣C)))
                        (新版)
```

表6.1に載せたものは,関数を定義するときにも使える関数でしたが,LISPにはシステムをコントロールするためだけに使われる関数があります。I/O版と新版では少々使い方が異なりますから,これを表6.2に示しておきます。

関数の定義

先にも述べましたようにLISPには関数しかありません。したがって,LISPですることといったら新たな関数を定義するか,関数の値を求めることしかありません。関数が自由に定義できるようになれば,もうLISPはわかったといってもいいでしょう。

以下に,関数の定義の仕方を説明します。

● I/O版の場合

DEFINEという関数を使います。その形式は,

```
(DEFINE␣関数名␣(LAMBDA␣引
数リスト␣関数の本体))
```

このうち,関数の本体は,引数リストの中の引数や,表6.1の関数を使って書かれたS式(リスト)です。たとえば,引数として与えられるリストの最後の要素を値とするような関数LASTは以下のように定義できます。

```
(DEFINE␣LAST␣(LAMBDA␣
(L)␣(COND␣(( NULL␣(CDR␣L
))␣(CAR␣L))
(T␣(LAST␣(CDR␣L)))))
```

なお,I/O版のLISPでは関数定義中に改行することができませんから,定義が終了するまで,文字を入力し続けます。

● 新版の場合

I/O版のDEFINEに対応するものとしては,DEという関数があります。DEによって関数を定義する場合の一般形は

```
(DE␣関数名␣引数リスト␣関数本体)
```

となっていて,DEFINEによるものよりも単純になっています。先のLASTはDEを使って,

```
(DE␣LAST␣(L)
 (COND␣((NULL␣(CDR␣L))
        (CAR␣L))
        (T␣(LAST␣(CDR␣L)))))
```

と定義することができます。この新版のほうでは,定義の途中で改行をすることができますから,見やすい状態でプログラムを作ることができます。

さらに,新版のほうにはDFという関数があります。これは,引数の数が決まっていない関数を定義するためのものです。その一般形は,次のようになっています。

```
(DF␣関数名␣引数リスト␣関数本体)
```

これは,形式的にはDEとまったく同じですが,引数リストの要素の個数が1個でなければならないという点が異なっています。そして,その要素は,実際にその関数を実行させるときの引数のリストを値としてもつようになっています。たとえば,

```
(DF WHAT(L)(……))
```

と定義されているWHATという関数を考えます。この関数の実行は,適当な個数の引数を付けて,

```
(WHAT␣3␣1␣4␣1)
```

などとします。このとき,引数であるLは

```
(3␣1␣4␣1)
```

というリストとして関数の本体に渡されていきます。このことを知っておけば,与えられた不定個の引数のうち,最大のものを値とするような関数MAXは以下のように定義できます。

```
(DE␣MAXH␣(L)
 (COND␣((NULL␣(CDR␣L))
        (CAR␣L))
       ((GREATERP
          (CAR␣L)
          (MAXH␣(CDR␣L)))
        (CAR␣L))
       (T␣(MAXH␣(CDR␣L)))))
```

ここでは,MAXはMAXHへ引数のリストを渡す役割しか果していませんが,EVALを使うと,MAXを直接に再帰定義をすることができます。

```
(DF␣MAX␣(L)
 (COND((NULL␣(CDR␣L))
       (CAR␣L))
      ((GREATERP␣(CAR␣L)
       (EVAL␣(CONS␣(QUOTE␣
       MAX)(CDR␣L)))␣(CAR␣L))
       (T(EVAL␣(CONS␣(QUOTE
       ␣MAX)(CDR␣L))))))
```

サンプルプログラム

リストを使った例題として,組み合わせに関するものを考えてみました。

問題　n人の人が集まって,プレゼントの交換をすることにしました。誰もが自分の持ってきたプレゼントをもらわないような組み合わせを,すべて求めるプログラムを作りなさい。

ひとつの表わし方として,

$(a_1\ a_2\ \cdots\ a_n)$　(a_i($i=1\sim n$)は$1\sim n$の数；$a_i \neq a_j$($i \neq j$のとき))

というリストを考えます。これは,i(i＝1～n)という人が,a_iという人が持ってきたプレゼントをもらうことを意味するものとします。たとえば,

```
(2␣3␣1)
```

は,1の人が2の人のプレゼントを,2の人が3の人のプレゼントを,3の人が1の人のプレゼントをもらうことを意味します。なお,このような順列を完全順列と呼ぶことがあるようです。また,考え方の概略を以下に示します。

(1)　n≦1なら,答えは存在しない。

(2)　n＝2なら,答えは(2␣1)のみ。

(3)　n≧3なら,次のふたつの場合がある。

(3.1) $(a_1\ a_2\ \cdots\ a_{n-1})$が(n－1)の場合ののひとつの答えであるとき,k＝1～(n

−1について，ai＝kであるようなiを探し，そこをnとして，aiを最後の要素にする。これはn人のときの答えの一部である。

(例)　(3␣1␣2)からは，(3␣4␣2␣1)　(3␣1␣4␣2)(4␣1␣2␣3)が得られる。

(3.2) k＝1～(n−1)について，ai＝kのものを探して，(3.1)と同じ置き換えをやるとき，i＝kであってもよい。このような(a_1␣a_2…␣a_{n-1})の列は次のようにして求められる。(a_1, a_2,…a_{n-2})が(n−2)人のときの答えのとき，ai＝k(k＝1～(n−2))となるものを探したら，aiを(n−1)とし，a_kをa_{n-1}にして，a_kをkにすればよい。

(例)　(2␣1)からは，(1␣3␣2)，(3␣2␣1)が得られ，そのうち，(1␣3␣2)からは，(4␣3␣2␣1)，(1␣3␣4␣2)，(1␣4␣2␣3)が得られる。

上の(1)～(3)を考えて作ったプログラムがリスト6です。そのうち，PP1, PP2がそれぞれ(3.1)(3.2)の手続きを行なうものです。ちなみに，n＝4のときの答えは以下のようになります。

(PP␣4)

(2␣3␣4␣1)(4␣3␣1␣2)(2␣4␣1␣3)(3␣4␣2␣1)(3␣1␣4␣2)(4␣1␣2␣3)(4␣3␣2␣1)(3␣4␣1␣2)(2␣1␣4␣3)

ちょっとひと言

LISPを使うときにメモリマップはあまり必要ではないと思われるので省略してあります。スタートアドレスは，I/O版が$3000，(コールドスタート)，新版が$C01Fまたは$C9B6(ホットスタート)です。それぞれ，ホット，コールドスタートは不明です。

ここで紹介したLISPは，実際のLISPのサブセットでしかありません。したがって，このLISPを使って人工知能を学ぶことは無理だと思われます。それよりも，LISPを使って，関数を再帰的に定義しながら，リストとは何か，LISPとは何かを学ぶほうがよいでしょう。そうするうちに，リスト構造が，複雑なデータを表わすのに，いかに便利なものであるかがわかってくると思います。

〈参考文献〉

- 中西正和　『Lisp入門(第二版)』　近代科学社，1981。
- 浅見俊幸他『LISP』I/O, 80年12月号。
- 渡辺卓也『PC-LISP』I/O, 81年6月号。
- 小倉哲『LISP』Oh！PC, 83年4月号。
- ADVANCED MZ-700　日本ソフトバンク出版部, 1984(予定)。

リスト6　プレゼント交換のやり方を求めるプログラム

```
(DE PP (N)
  (COND
    ((LESSP N 2) NIL)
    ((EQUAL N 2) (QUOTE ((2 1))))
    (T (APPEND
         (PP1 (PP (SUB1 N)) N)
         (PP2 (PP (SUB1 (SUB1 N)))
                  (SUB1 N))))
  ))


(DE PP1 (L N)
  (COND
    ((NULL L) L)
    (T (APPEND
         (SWP  (CAR L) 1 N)
         (PP1 (CDR L) N) ))
  ))


(DE SWP (L M N)
  (COND
    ((GREATERP M (SUB1 N)) NIL)
    (T (CONS (SWP1 L M N)
             (SWP L (ADD1 M) N) ))
  ))


(DE SWP1 (L M N)
  (COND
    ((EQUAL (CAR L) M)
        (APPEND (CONS N (CDR L))
                (LIST M) ))
    (T (CONS (CAR L)
             (SWP1 (CDR L) M N) ))
  ))


(DE PP2 (L N)
  (COND
    ((NULL L) L)
    (T (APPEND
         (SWP2 (CAR L) 1 N)
         (PP2 (CDR L) N) ))
  ))
(DE SWP2 (L M N)
  (COND
    ((GREATERP M (SUB1 N))
        (LIST
          (APPEND L
                  (LIST (ADD1 N) M) )
    ))
    (T (CONS
         (MAK M (ADD1 N)
             (APPEND (SWP3 L M N)
                 (LIST (ITEM M L) M)))
         (SWP2 L (ADD1 M) N) ))
  ))

(DE SWP3 (L M N)
  (COND
    ((EQUAL (CAR L) M)
        (CONS N (CDR L)) )
    (T (CONS (CAR L)
             (SWP3 (CDR L) M N) ))
  ))

(DE MAK (M N L)
  (COND
    ((ONEP M) (CONS N (CDR L)))
    (T (CONS (CAR L)
             (MAK (SUB1 M) N (CDR L))
    ))
  ))

(DE ITEM (N L)
  (COND
    ((ONEP N) (CAR L))
    (T (ITEM (SUB1 N) (CDR L)))
  ))
```

FORTH

(M-FORTHはモニタ変換しないとセーブ・ロードができない)

概　要

FORTHはアメリカの天文学者C.H.Moore氏によって，天体望遠鏡などの天文装置を制御するためのプログラミング言語として考案されと言われています。FORTHとは，『第4』を意味するfourthを縮めたものです。これには，真空管，トランジスタ，LSIに続く第4世代のコンピュータでの主要言語になってほしいという強い願いが込められているのではないでしょうか。

FORTHも，LISPなどと同じく，一部に狂信的ともいえる愛好者がいるわりに，それほど有名ではありません。しかし，FORTHを実際に使ってみれば，確かにこれは素晴らしい言葉だということがわかるでしょう。

MZで使えるFORTHには，テープアスキーから販売されているFORTH/Σと，テクノソフトから販売されているM（ミディアム)-FORTHがあります。後者のM-FORTHの方は，I/O誌の81年3月号に発表されましたが，FORTH/Σのほうは，MZ用のものは発表されていません。しかし，ASCII誌に82年4月号から，PC80版の解説が載っていますからそれを見ればよいでしょう(MZ-80用のFORTH/Σのマニュアルは，その連載の第1回目と2回目をそのまま転載したものです)。

FORTHは，その演算の基本をスタックにおいています。スタックとは，データを格納しておく棚のことで，あとから置いたものを先に取り出すという性質があります。つまり，次の演算で使用するデータは，常にスタックのトップ（1番目）にあることがわかってますから，能率がよくて高速な演算をすることができます。

FORTHを使用する上での最大の特色は，その対話性の良さです。利用者は，電卓を扱うのと同じぐらいの気楽さでプログラムを実行させることができます。FORTHでは命令のことをワードと呼んでいます。そして，プログラムするとは，すでにあるワードから新たなワードを作ることに他なりません。これは，LISPにおいて，関数を定義することがプログラムであったのと似ています。しかし，LISPがプログラムをリストの形で保存しておいたのに対し，FORTHは，新たなワードをその場でコンパイルしてしまう点が異なっています。このため，プログ

ラムの実行速度が高速になります。

スタックと逆ポーランド記法

FORTHでは，演算はスタックを介して行なわれます。したがって，利用者はスタックの状態を，常に頭に入れてなければなりません。なお，FORTHには，データを蓄えておくためのパラメータスタックと，ワードを呼ぶときの帰り番地を蓄えるためのリターンスタックがあります。単にスタックと言う場合は，パラメータスタックのことを言っているものと思ってください。

FORTHのワードは，すべてがスタックに対する操作を行なうものです。たとえば，

10

と書く場合，他の言語では，それがデータであること以外には何の意味もありません。しかし，FORTHでは，これも立派な命令で，「スタックのトップに10を積む」ことを意味します。

また，スタックを演算の基本としているため，命令の記述は逆ポーランド記法によってなされます。逆ポーランド記法では，

3＋5

などという数式は，

3　5　＋

という具合に，演算子を最後にもってくる形で表記されます。偶然にも，この表記は日本語に近くなっています。上の例は，

「3に5を加える」

と読むことができます。同様に，

（3＋5）＊7＋2＊4

という数式は，逆ポーランド記法で，

3　5　＋　7　＊　2　4　＊　＋

となり，これは，

「3と5を加えて，7をかけ，それに，2と4をかけたものを加える」

と読むことができます。

FORTHでは，逆ポーランド記法で書かれた命令を，ひとつずつスタックに対して行なっているにすぎません。

＋と＊が，それぞれ，スタックのトップと2番目を，加えたりかけた値をスタックに積むワードであることを知っていれば，

3　5　＋　7　＊　2　4　＊　＋

を実行するとき，スタックがどのように変化するのかを知るのは簡単でしょう。このときのスタックの変化を図7.1に示しておきます。

ワードの定義

FORTH/ΣやM-FORTHで使用可能なワードを表7.1に示しておきます。

FORTHで新たなワードを定義する場合は表7.1(7)の定義用ワードを用いればよいわけですが，ここで，FORTH/Σを使った例を示します。

新たなワードを定義するためのワードは，：と；(M-FORTHでは[と])です。これを，

：␣名前␣ワードや数の列␣；

という形で用いれば，その内容がコンパイルされて，新しいワードが登録されます。たとえば，スタックのトップの2乗をスタックに積むためのワードSQRは，

：␣SQR␣DUP␣＊␣；

と定義できます。

なお，FORTHでは，ワードとワードの区切りを空白で識別します。このため，ワードとワードの間には，ひとつ以上の空白を入れなければなりません。名前が空白であるようなワードを定義することもできません。

FORTHは，PASCALなどの構造化プログラミング言語よりも，ずっとプログラムを構造化するのにむいています。FORTHのむちゃくちゃともいえる構文（プログラムのどの部分を取り出しても，その部分が何かの実行を行なう）は，プログラムをモジュール化するのに便利です。

したがって，FORTHにおける一般的なプログラミングの態度は，目的に必要な手段を想定し，手段の積み重ねで目的を満たすプログラムを作るという，ボトムアップなものになります。このため，使用するワードは，すべて，定義されている必要があります。

さて，プログラムを構造化するためには，条件分岐と繰り返しをするためのワードが必要です。そのために，よく使われるワードは，IF～(ELSE)～THEN，BEGIN～END，DO～LOOPです。それぞれは，PASCALでのIF文，REPEAT文，FOR文に対応しています。これらのワードの働きは，FORTHで新しいワードを定義するために，ぜひとも理解しておかねばなりません。それらの働きを文で説明してもわかりづらいので，制御の流れを図7.2の流れ図に示しておきます。な

図7.1　3␣5␣＋␣7␣＊␣2␣4␣＊␣＋を実行するときのスタックの変化

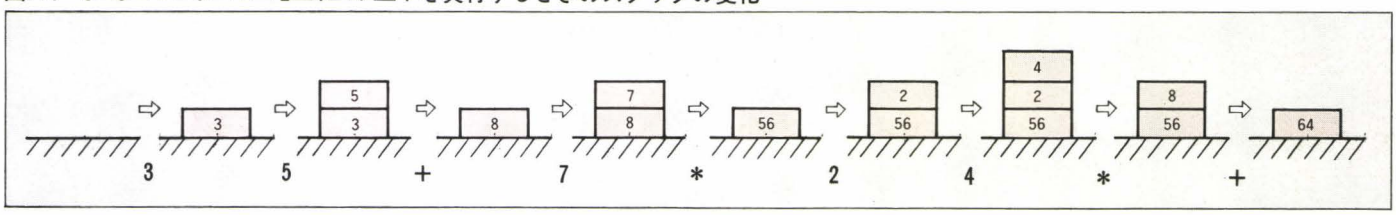
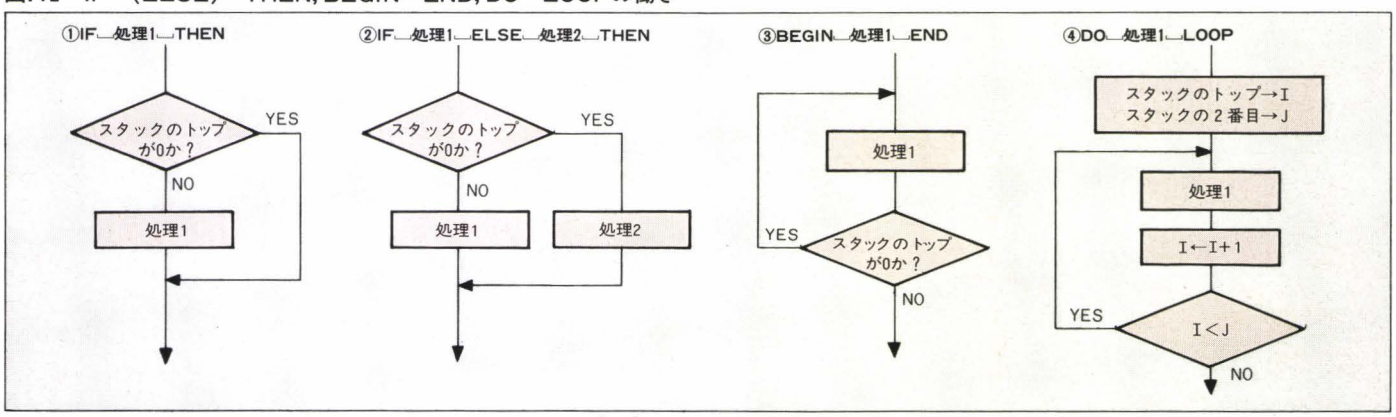


図7.2　IF～(ELSE)～THEN，BEGIN～END，DO～LOOPの働き



（注）　①～③で条件判断を1回するごとに，スタックのトップが取り去られる。
④でのI，Jは便宜的な変数。実際には，スタックのトップと2番目のデータはリターンスタックに積まれる。

お，これら制御用のワードは：と；(M-FORTHでは[と])による定義の中でしか用いることはできません。次の例は，スタックのトップが0でないときは，0～9を出力し，−1を出力するワードEXの定義です。なお，Iとは，ループカウンタの値をスタックに積むためのワードです。

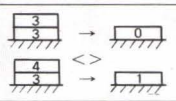
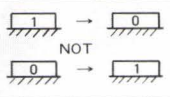
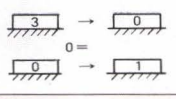
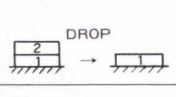
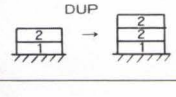
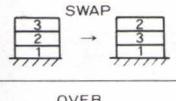
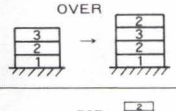
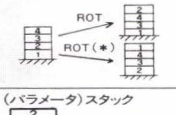
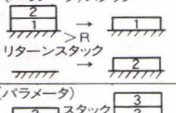
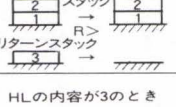
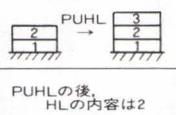
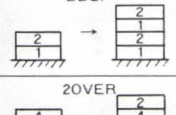
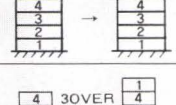
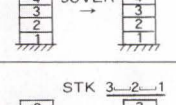
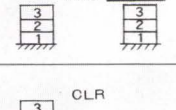
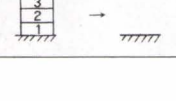
```
:␣EX␣0␣=␣IF␣−1␣.␣ELSE
10␣0␣DO␣I␣.␣LOOP␣THEN;
```

表7.1　FORTHのワード　ワード名が上がFORTH/Σ，下がM-FORTH

ワード名	スタック変化	意　味
(1)出力用ワード(例ではHEXと書いてあるものの基数が16，他は10)		
. .	10␣ ・ 10 10 →	スタックのトップをそのときの基数(BASE)で出力する。
TYPE S.	"ABC"␣ S. ABC アドレス / 3 →	スタックのトップを文字列が格納されている先頭のアドレス，スタックの2番目を文字数とみて，文字列を出力する。
CR LF		改行と復帰を行なう。
C. ／	300␣C.　44 (＄012C)　(＄2C) 300 →	スタックのトップの下位8ビットを，そのときの基数で出力する。
? ／	＄8000 OA ＄8001 EF　のとき HEX␣8000 ? EF0A ＄8000 →	スタックのトップをアドレスとみなして，その内容を出力する。
SPACE ／		スペースを出力する。
SPCES ／	3␣SPCES 3 →	スタックのトップの個数だけ，スペースを出力する。
ECHO ／	HEX␣1241␣ A ECHO ＄1241 →	スタックのトップの下位8ビットをASCIIコードとみなして，それに対応する文字を出力する。
／ #	−1␣# 65535 −1 →	スタックのトップを符号なしの10進数とみなして出力する。
／ ＄	−1␣＄. FFFF −1 →	スタックのトップを符号なしの16進数とみなして出力する。
(2)算術演算用ワード(例では，基数はすべて10)		
+ +	a, b → a+b	スタックのトップと2番目を加えて，スタックに積む。
− −	a, b → a−b	スタックの2番目からトップを引いて，スタックに積む。
* *	a, b → a×b	スタックのトップと2番目をかけて，スタックに積む。
／ ／	a÷bがq余りrのとき a, b → q	スタックの2番目をスタックのトップで割り，その商をスタックに積む。
／MOD ／M	a÷bがq余りrのとき ／MOD a, b → q, r	スタックの2番目をスタックのトップで割り，商をトップに，余りを2番目に積む。
MOD ／	a÷bがq余りrのとき MOD a, b → r	スタックの2番目をスタックのトップで割り，余りをスタックに積む。

ワード名	スタック変化	意　味
ABS ／	ABS −3 → 3	スタックのトップの絶対値を求め，スタックに積む。
MINUS ／	MINUS 3 → −3	スタックのトップの2の補数を求め，スタックに積む(符号反転)。
MAX ／	MAX 5, 3 → 5	スタックのトップと2番目のうち，大きいほうをスタックに積む。
MIN ／	MIN 5, 3 → 3	スタックのトップと2番目のうち，小さいほうをスタックに積む。
(3)論理演算(ビットごと)用ワード(例では基数が16)		
AND AND	AND ABCD, 0FFF → 0BCD	スタックのトップと2番目の，ビットごとの論理積を求め，スタックに積む。
OR OR	OR ABCD, 0FFF → AFFF	スタックのトップと2番目の，ビットごとの論理和を求め，スタックに積む。
XOR XOR	XOR F0F0, 00FF → F00F	スタックのトップと2番目の，ビットごとの排他的論理和を求め，スタックに積む。
COM ／	COM 0F0F → F0F0	スタックのトップの1の補数を求め，スタックに積む(ビットの反転)。
NAND ／	NAND ABCD, 0FFF → F432	スタックのトップと2番目の，ビットごとの論理積を求め，それをビット反転したものをスタックに積む。
NOR ／	NOR ABCD, 0FFF → 5000	スタックのトップと2番目の，ビットごとの論理和を求め，それをビット反転したものをスタックに積む。
SWAB ／	SWAB ABCD → CDAB	スタックのトップの上位バイトと下位バイトを変換したものをスタックに積む。
(4)比較用ワード		
= =	3, 3 → 1 = 4, 3 → 0	スタックのトップと2番目が等しいときは真(1)を，そうでないときは偽(0)をスタックに積む。
< <	3, 4 → 0 < 4, 3 → 1	スタックの2番目がトップより小さいときは真(1)を，そうでないときは偽(0)をスタックに積む。
> >	3, 4 → 1 > 4, 3 → 0	スタックの2番目がトップより大きいときは真(1)を，そうでないときは偽(0)をスタックに積む。
<= ／	3, 4 → 0 <= 4, 3 → 1	スタックの2番目がトップより小さいか等しいときは真(1)を，そうでないときは偽(0)をスタックに積む。
>= ／	3, 4 → 1 >= 4, 3 → 0	スタックの2番目がトップより大きいか等しいときは真(1)を，そうでないときは偽(0)をスタックに積む。

表7.1 （つづき）ワード名は上がFORTH/Σ，下がM-FORTH

ワード名	スタック変化	意　味
<>		スタックの2番目とトップが等しくないときは真(1)を，等しいときは偽(0)をスタックに積む。
NOT		スタックのトップが偽(0)なら真(1)を，そうでなければ，偽(0)をスタックに積む。
0=		スタックのトップが0なら真(1)を，そうでなければ，偽(0)をスタックに積む。NOTと同じ。
(5)スタック操作用ワード		
DROP DROP		スタックのトップを取り去る。
DUP DUP		スタックのトップを複製して，スタックに積む。
SWAP SWAP		スタックのトップと，2番目を交換する。
OVER OVER		スタックの2番目を複製して，スタックに積む。
ROT ROT(*)		スタックのトップから3番目を回転させて，3番目をトップにもってくる。(*)M-FORTHでは，スタック全体で回転を行なう。
>R		(パラメータ)スタックのトップを，DO～LOOPで使用されるリターンスタックに積む。
R>		リターンスタックのトップを，(パラメータ)スタックに積む。
PUHL		HLレジスタの内容をスタックに積む。H,Lレジスタの内容がそれぞれ上位，下位バイト。
POHL		スタックのトップをHLレジスタに入れる。
DDUP		スタックのトップと2番目を複製してスタックに積む。OVER OVERと同じ。
2OVER		スタックの3番目を複製して，スタックに積む。
3OVER		スタックの4番目を複製して，スタックに積む。
STK		スタックの内容を表示する。スタックの内容は変わらない。
CLR		スタックを空にする。

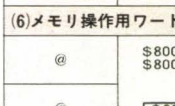
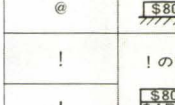
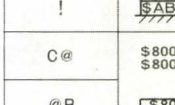
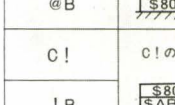
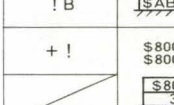
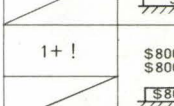
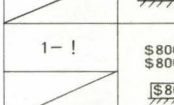
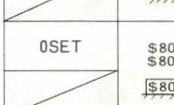
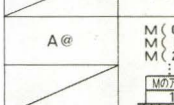
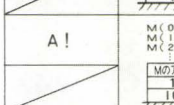
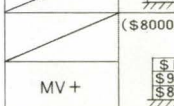
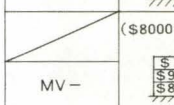
ワード名	スタック変化	意　味
(6)メモリ操作用ワード		
@ @		スタックのトップをアドレスとみなして，そのアドレスの内容(2バイト)をスタックに積む。
! !		スタックのトップをアドレスとみなして，2番目を，そのアドレスへ転送する。
C@ @B		@で1バイトのみスタックへ積む。上位バイトは0。
C! !B		!でスタックの2番目の下位バイトのみ転送する。
+! 		スタックのトップをアドレスとみなして，そのアドレスの内容にスタックの2番目を加える。
1+! 		+!で，スタックの2番目の代りに1を加える。
1-! 		+!で，スタックの2番目の代りに－1を加える。
0SET 		スタックのトップをアドレスとみなし，そのアドレスと，ひとつ先のアドレスの内容を0にする。
A@ 		スタックのトップを配列の先頭アドレス，2番目を添字とみなして，配列の値をスタックに積む。
A! 		スタックのトップを配列の先頭アドレス，2番目を添字とみなして，3番目で与えられるデータを，その位置へ格納する。
 MV+		スタックのトップのバイト数だけ，3番目から2番目のアドレスへ，アドレスを増加させながら，メモリの内容をブロック転送する。
 MV-		スタックのトップのバイト数だけ，3番目から2番目のアドレスへ，アドレスを減少させながら，メモリの内容をブロック転送する。
(7)定義用ワード		
: ; [ ]	スタックのトップと2番目の和を出力するワードADDの定義 :␣ADD␣+␣␣;	新たなワードを定義する。一般形は :␣名前␣すでに定義されているワードや数の列␣; [␣名前␣すでに定義されているワードや数の列␣]
MCODE NEXT M[ ]M	スタックのトップの下位バイトをプリンタへ出力するワードLPの定義　MCODE␣LP␣POH␣7D␣CD␣8F␣01␣NEXT	機械語によるワードを定義する。一般形は， MCODE␣名前␣ワードや16進数(1バイト)の列␣NEXT M[␣名前␣1バイトで表わせる数の列␣]M RET(C9)は書かない。
CONST CONST	1という値を持つ変数ONEを定義する。 1␣CONST␣ONE	定数を定義する。一般形は 数値␣CONST␣定数名
VAR VAR(*)	Xという変数を宣言する VAR␣X	変数を宣言する。一般形は VAR␣変数名 (*)M-FORTHでは，スタックのトップの値が変数の初期値になる。
<BUILDS DOES> 	2次元配列を宣言するためのワードDIMを定義する。 :␣DIM␣<BUILDS␣2␣*␣ *␣ALLOT␣DOES>␣;	定義用のワードを定義する。一般形は :␣名前␣<BUILDS␣ワードや数の列␣DOES>␣;

表7.1 （つづき）ワード名は上がFORTH/Σ，下がFORTH

ワード名	例	意　味
ARRAY ／	大きさが10(20バイト)の1次元配列Aを宣言する。 10␣ARRAY␣A	1次元配列を宣言する。一般形は 大きさ␣ARRAY␣配列名
[ ] ／	:␣X␣[␣HEX␣]␣F␣[␣DEC␣]␣10␣+␣; は，基数が10のときの :␣X␣15␣10␣+␣; と同じ。	:定義の途中で，一時的に[ ]内のワードを実行させる。
( ) ／	:␣X␣*␣(␣KAKERU)␣;	(　と　)で囲まれた文字列はコメントとなる。 一般形は (␣文字列) となり，最初の(と文字列の間には空白をおく。
／ " "	"ABC" → アドレス 3	" と " の間が文字列であることを示す。このとき，文字列の格納開始アドレスと，文字数がスタックに積まれる。
(8)制御用のワード(ワードの定義中にのみ使用できる)		
IF (ELSE) THEN IF (ELSE) THEN	スタックのトップが0ならFALSE，0以外ならTRUEを出力するワードQを定義する。 :␣Q␣0␣=␣IF␣"FALSE"␣ELSE␣"TRUE"␣THEN␣TYPE␣;	IF␣スタックのトップが真（0以外）のときの処理␣ELSE␣スタックのトップが偽(0)のときの処理␣THENの形で使う。ELSE以下が必要ないときはELSEは省く。
BEGIN END BEGIN END	9から1までを出力するワードDを定義する :␣D␣9␣BEGIN␣DUP␣.␣1␣−␣DUP␣NOT␣END␣DROP␣;	ENDの実行時にスタックのトップが真（0以外）になるまでBEGINとENDの間の処理を繰り返す。
DO (LEAVE) LOOP DO LOOP	0から9までを出力するワードPを定義する。 :␣P␣10␣0␣DO␣I␣.␣LOOP␣;	(パラメータ)スタックのトップ，2番目を，それぞれ初期値，終値としてリターンスタックに積み，LOOPが実行されるごとにカウンタを1だけます。カウンタの値が終値以上になるまでDOとLOOPの間を繰り返す。LEAVEは，カウンタの値と終値と等しくする機能がある。
I I		DO～LOOP実行時のカウンタの値を（パラメータ）スタックへ積む。

BEGIN
WHILE
REPEAT
／
	スタックのトップ−1だけの個数の*を出力するワードSTの定義。
	:␣ST␣BEGIN␣1␣−␣DUP␣WHILE␣42␣ECHO␣REPEAT␣;
		BEGIN␣処理␣WHILE␣スタックのトップが真(0以外)のときの処理␣REPEAT
		の形で使う。REPEATが実行されると，制御は，BEGINの直後に移る。

DO (LEAVE) +LOOP ／	0から8までを2ずつ増加させて，プリントするワードP2を定義する。 :␣P2␣10␣0␣DO␣I␣.␣2␣+LOOP␣;	DO～LOOPで，カウンタの増減を1ではなく，スタックのトップだけカウンタを増加させるもの。
／ F.IF (ELSE) THEN		F.IF␣スタックのトップが偽(0)のときの処理␣ELSE␣スタックのトップが真(0以外)のときの処理␣THEN の形で使う。
／ J		DO～LOOP実行時のカウンタのとるべき終値をスタックに積む。
(9)辞書操作用ワード		
FORGET FORGET		FORGET␣ワード名 の形で使う。指定されたワード以後に定義されたワードをすべて消去する。
HERE ／		次にワードが登録されるべきアドレス(辞書ポインタの内容)をスタックに積む。
ALLOT ／	VAR␣X␣10␣ALLOT Xという名前で示されるアドレスから10ワード分確保する。	スタックのトップの値だけ辞書ポインタを進める。 作業用の領域を確保するのに使用する。
(コンマ) ／		スタックのトップを辞書ポインタの示すアドレスと，そのアドレスの次のアドレスに格納する。辞書ポインタは2進む。
C, ／		スタックのトップの下位バイトを辞書ポインタの示すアドレスに格納する。辞書ポインタは1進む。

ワード名	例	意　味
(アポストロフィ) ／	HEX␣'␣FORGET とすると，FORGETの先頭番地($2B58)がスタックに積まれる。	'の次にくるワードや変数などの先頭アドレスをスタックに積む。このアドレスから，実際の機械語が書かれている。
／ (コンマ)		辞書ポインタの示すアドレスから2バイト確保し，辞書ポインタを1進める。 FORTH/Σの1␣ALLOTと同じ。
(10)その他のワード		
BASE BASE	8␣BASE␣! 以後の入力を8進数によるものとする。	基数(8, 10, 16)が格納されているアドレス。
EDIT TXT		テキストエディタへ制御を移す。
LOAD LOAD		カセットテープから，エディタで作られたソースプログラムをロードする。
SAVE SAVE		カセットテープに，エディタで作ったプログラムをセーブする。
USR USR(*)		スタックのトップを機械語サブルーチンの先頭アドレスとみなして，そこへジャンプする。 (*)M-FORTHではREGという名前のエリアを介してレジスタの値の受け渡しができる。 REGの配置　F A C B E D L H
VERIFY VERI		カセットテープのベリファイをする。
INKEY GETKY		キーが押されていれば，そのASCIIコードを，押されていなければ0をスタックに積む。
INPUT ?(*)		数値入力があるまで待ち，その値をスタックに積む。 (*)M-FORTHでは文字でもよい。
(10)その他のワード(FORTH/Σのみ)		
HEX	HEX␣15␣26␣+␣. 003B	基数を16にする。すなわち，数を16進数とみなす。
DEC.	DEC␣15␣26␣+␣. 41	基数を16にする。すなわち，数を10進数とみなす。
OCT	OCT␣15␣26␣+␣. 000043	基数を8にする。すなわち，数は8進数とみなす。
LIST		現在までに登録されたワードを逆コンパイルして表示する。
EXEC		エディットバッファの内容(エディタで作ったプログラム)を実行する。
CLB		エディットバッファを，メモリとして使えるようにする。
BYE		モニタへジャンプする。
LOCATE.	10␣10␣LOCATE␣42␣ECHO (10, 10)の位置に*を書く。	スタックの2番目，トップを，それぞれ，x, y座標とみなして，その位置にカーソルを移動する。
SCREEN		スタックの2番目，トップを，それぞれ，x, y座標とみなして，その位置にある文字のディスプレイコードをスタックに積む。
LPRINT		文字の出力を画面からプリンタへ切り換える。
CRT		文字の出力をプリンタから画面へ切り換える。
SFORTH		FORTHをそれ自身，最近のワードまでを含めてカセットテープにセーブする。

表7.1 (つづき)

ワード名		意　　味
⑩その他のワード(M-FORTHのみ)		
CORDR		スタックをトップから順にz, y, x とすると, zを0.5°単位として, $x\cos z - y\sin z$を2番目に, $x\sin z + y\cos z$をトップに積む。
CORDV		スタックをトップから順にz, y, x, とすると, zを0.5°単位として, $\sqrt{x^2+y^2}$ を2番目に, $z + \tan^{-1}\frac{y}{x}$をトップに積む。
SETG		スタックの2番目, トップをx, y座標とみなして, その位置のドットをセットする。
RESG		スタックの2番目, トップをx, y座標とみなして, その位置のドットをリセットする。
LINE	0␣0␣10␣20␣LINE (0, 0)から(10, 20)へ直線を引く	スタックをトップからy_2, x_2, y_1, x_1とすると, 点(x_1, y_1)から(x_2, y_2)へ直線を引く。
VRCL		仮想VRAMをクリアし, 仮想VRAM のスイッチをONにする。
VRMV		仮想VRAMのスイッチがONなら, 仮想VRAMの内容をVRAMに移し, 仮想VRAMのスイッチをOFFにする。
VRCH		仮想VRAMのスイッチを切り換える。仮想VRAMがONなら, SETG, RESG, LINEは仮想VRAM上に行なわれるので画面には現われない。
LIMIT	a␣b␣LIMIT パラメータスタック, データスタックのボトムアドレスを, a, bとする。	パラメータスタック, リターンスタックのBOTTOMアドレス(空のとき指しているアドレス)を変更する。
APLY	3␣4␣*+"␣APLYは 3␣4␣+と同じ。	スタックのトップと2番目で表わされる文字列(*の項参照)と, 同じワードを実行する。
SP		リターンスタックのBOTTOMアドレスが格納されているアドレス。
DS		パラメータのスタックのBOTTOMアドレスが格納されているアドレス。
PE		辞書ポインタ。

表7.2　エディタコマンド(主なもののみ)

コマンド	意　　味
A	エディットモードより抜けてFORTHへ戻る。
!	
Tn	ラインポインタの指す行からn行分表示する。 ラインポインタは変化しない。
Tn	
L±n	ラインポインタをn行先に進める(+n)か, n行戻す(−n)。
±n	
Is	ラインポインタの指す行の前へ, sというテキストを挿入する。ラインポインタは変化しない。
I CR S	
n	ラインポインタを先頭からn行目にセットする。
Pn	
Kn	ラインポインタの指す行からn行削除する。
Dn	
ESf	ソーステキストをfというファイル名でカセットテープにセーブする。
wf	
ELf	カセットテープのソーステキストをロードする。
L	
	ソーステキストを実行する(FORTH/Σでは, 実行モードでEXECがある)。
F	
P	ラインポインタの指す行をエディットする。
␣S	ラインポインタの指す行の後へ, Sというテキストを挿入し, ラインポインタをその行に移す。

上側がFORTH/Σ, 下側がM-FORTHのコマンド

エディタ

FORTHのプログラミングは対話形式で行なうことが基本ですが, 長いプログラムを作る場合など, エディタを使用することができます。エディタを使用すれば, プログラムの一部だけを修正できるので, エラーがあった場合, ワードの定義を最初からやり直す手間を省くことができます。表7.2にエディタコマンドの一覧表を示します。FORTH/Σもも M-FORTHもラインポインタによってテキストを管理するラインエディタです。どちらのエディタも似かよった機能をもっていますが, FORTH/Σには, Pコマンドという, 行の内容をエディットするコマンドがあります。これについて少し説明します。いまラインポインタの指す行が

␣:␣WDW?␣ADAYS

であるとします。このとき,

P／／／／／!／B CR

と入力すると, ／以外の文字を書いたところが置き換わって,

␣:␣WDW!␣BDAYS

となります。また, この行に対して

P／／／／ INST OO CR

とすると, INST を押した部分に対応する位置にそれ以後の文字列が挿入されて,

␣:␣WDOOW!␣BDAYS

となります。 CLR を使えば文字列を消去することができます。上の例に対して,

P／／／／ CLR ␣,␣,␣ CR

とすると, CLR が押された部分に対応する位置から, そのあとに入力した文字数だけが消去されて,

␣:␣WD!␣BDAYS

となります。

このように, Pコマンドは, 現在入力したばかりの行をすぐ修正するのによく使われます。

リスト7　カレンダーをプリンタへ出力するプログラム

```
1:      MCODE LPOUT
2:       POHL 7D CD 8F 01
3:      NEXT
4:      DEC
5:      12 ARRAY MONTH
6:      : URU? 4 MOD NOT ;
7:      : M! MONTH A! ;
8:      : M@ MONTH A@ ;
9:      : YEARS
10:        0 SWAP DUP 1984 <> IF 1984
11:        DO I URU? IF 366 +
12:             ELSE 365 + THEN LOOP
13:        ELSE DROP THEN
14:     ;
15:     : INIT
16:        31 12 0 DO DUP I M! LOOP
17:       1 -  3 M!  30  5 M!
18:        30  8 M!  30 10 M!
19:     ;
20:     : USET
21:        URU? IF 29 1 M!
22:             ELSE 28 1 M! THEN
23:     ;
24:     : MDAYS
25:        0 SWAP DUP 1 <> IF
26:        1 DO I 1 - M@ + LOOP
27:        ELSE DROP THEN
28:     ;
29:     : DAYS SWAP MDAYS + 1 - ;
30:     : ADAYS
31:        INIT ROT DUP USET YEARS
32:        ROT ROT DAYS +
33:     ;
34:     : SPOUT 32  LPOUT ;
35:     : CRLF 13 LPOUT ;
36:     : DAY SPOUT SPOUT DUP LPOUT ;
37:     : TITL2
38:        112 CRLF 7 0 DO DAY 1 +
39:        LOOP CRLF DROP
40:     ;
41:     : WDW? ADAYS 7 MOD ;
42:     : LC 11 LPOUT ;
43:     : SC 12 LPOUT ;
44:     : SPCS 0 DO SPOUT LOOP ;
45:     : SPS
46:        10 >= IF 1 ELSE 2 THEN SPCS
47:     ;
48:     : COUT
49:        10 /MOD DUP 0 > IF
50:        48 + LPOUT ELSE DROP THEN
51:        48 + LPOUT
52:     ;
53:     : I3 DUP SPS COUT ;
54:     : PRDAY
55:        SWAP DUP 0 > IF DUP 3 *
56:        SPCS THEN 1 - SWAP 1 + 1
57:        DO DUP I + 7 MOD 0 = IF
58:        I 1 > IF CRLF THEN THEN
59:        I I3 LOOP CRLF DROP
60:     ;
61:     : TITL1 LC I3 SC TITL2 ;
62:     : CALM
63:        DUP TITL1 DUP ROT SWAP
64:        1 WDW? SWAP 1 - M@ PRDAY
65:     ;
66:     : CALY
67:        1 + SWAP DO DUP I CRLF CRLF
68:        CALM LOOP
69:     ;
```

サンプルプログラム

FORTH/Σのエディタを使って，カレンダーをプリンタへ出力するプログラムを作ってみました。これをリスト7に示します。このプログラムは，1984年以降の特定の月のカレンダを出力するプログラムCALMが核になっています。ある特定の月から特定の月までのカレンダを出力するプログラムCALYは，CALMをDO～LOOPで繰り返したものに過ぎません。このプログラムの使用法は，プログラムをコンパイルしたあと，実行モードで，

1998␣3␣CALM　[1998年3月のカレンダ]

2001␣4␣6␣CALY [2001年4月から6月のカレンダ]

などとします。図7.3は，

1984␣3␣7␣CALY

としたときの実行結果です（確かに合っている）。

なお，プログラム中，WDW？はある特定の年月日が何曜であるかを求めるものです（年␣月␣日␣WDW？で使用するとスタックのトップに0～6の値が積まれる。0が日曜）が，カレンダを書くためには，指定する月の1日が何曜であるかがわかればよいので，この点，少し無駄といえます。

図7.3　1984年3月から7月のカレンダー

```
    3
 日 月 火 水 木 金 土
              1  2  3
 4  5  6  7  8  9 10
11 12 13 14 15 16 17
18 19 20 21 22 23 24
25 26 27 28 29 30 31

    4
 日 月 火 水 木 金 土
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30

    5
 日 月 火 水 木 金 土
        1  2  3  4  5
 6  7  8  9 10 11 12
13 14 15 16 17 18 19
20 21 22 23 24 25 26
27 28 29 30 31

    6
 日 月 火 水 木 金 土
                 1  2
 3  4  5  6  7  8  9
10 11 12 13 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30

    7
 日 月 火 水 木 金 土
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30 31
```

ちょっとひと言

FORTHもLISPと同じく，代入文のない言語です。この点は，今はやりのLOGOについても言えることです。したがって，BASICを知っている人がFORTRANやPASCALを学びやすいように，LOGOやLISPを知っている人がFORTHを学ぶのは簡単でしょう。LISPにしろFORTHにしろ，BASICとはひと味違った世界が，そこにはありますから，一度ぐらいは使用してみるのがいいでしょう。きっと病みつきになるかもしれません。

なお，書き忘れましたが，FORTH/Σ，M-FORTHのスタート番地は，どちらも$1200(コールドスタート)ですが，このとき，今までに登録されているワードは消去されません。M-FORTHにはホットスタートもあって，それは$1268か$1925です。

〈参考文献〉

- 大園富美男他『M-FORTH/MZ』I/O，81年3月号。
- MZ-80　FORTH-Σ　マニュアル
- 原道宏『FORTH』トランジスタ技術，81年2月号～3月号。
- 片桐明　『プログラム表記から見た言語比較』information，　84年3月号。

言語の比較

それぞれの言語によるソート

これまでに，GAME，PALL，FORM，μPASCAL，TL/1，LISP，FORTHとクリーンコンピュータのための言語の紹介をしてきました。それぞれの言語には得意な分野とそうでない分野があります。しかし，どれも同じプログラミング言語ですから，ちょっと工夫すれば，何でもやらせることができると思われます。そこで，上記の言語を使って，同一の目的のためのプログラムを作ってみました。目的は，数字のSORT(整列)です。SORTにはいろいろのやり方がありますが，ここでは次のようなやり方をしています。

$a_m, a_{m+1}, \cdots, a_{n-1}, a_n$ が大きい順に並んでいるとき，a_{m-1}を，それを挿入しても大小関係が乱れない位置に挿入します。たとえば，$a_{m-1} < a_i$, $a_m \geqq a_{i+1}$ならば，a_{m-1}を挿入した結果は，

$$a_m, a_{m+1}, \cdots, a_i, a_{m-1}, a_{i+1}, \cdots, a_n$$

となります。これを新たに，

$$a_{m-1}, a_m, \cdots, a_{n-1}, a_n$$

とみなして，a_{m-2}を挿入します。これを，挿入する要素がなくなるまで繰り返します。

この方法でSORTを行なうプログラムがリスト8～リスト15に示してあります。SORTすべき要素は，乱数を使ったもの（GA

リスト8　GAMEによるソート

```
1000 L=$8000
1010 N=14
1020 I=0,N
1030  L:I)='9
1040 @=I+1
1050 ?=4000
1060 ?=2000
1070 /
1080 ?=4000
1090 #=-1
2000 J=1,N
2010  P=N-J
2020  X=L:P)
2030  ?=3000
2040 @=J+1
2050 ]
3000 K=P+1
3010 B=(X<L:K)*(K<=N)
3020 ;=B=0 #=3060
3030 L:K-1)=L:K)
3040 K=K+1
3050 #=3010
3060 L:K-1)=X
3070 ]
4000 A=0,N
4010  ?=L:A) .=1
4020 @=A+1
4030 ]
```

リスト9　PALLによるソート

```
ARRAY L[20];
VAR I,N;
PROCEDURE INST(X,S,E);
 BEGIN
  IF (S>E)
   THEN
    L[S-1]←X
   ELSE
    IF (X<L[S])
     THEN
      BEGIN
       L[S-1]←L[S];
       INST(X,S+1,E);
      END
     ELSE
      L[S-1]←X;
 END;
PROCEDURE SORT(S,I,E);
 BEGIN
  IF (S<I) THEN
   BEGIN
    INST(L[I-1],I,E);
    SORT(S,I-1,E);
   END;
 END;
PROCEDURE PRINL(S,E);
 VAR I;
 BEGIN
 I←S;
 E←E+1;
 REPEAT
  WRITE(L[I]+48,$20);
 UNTIL INC(I)=E;
 END;
BEGIN
 N←9;
 L[0]←3;L[1]←1;L[2]←4;L[3]←1;L[4]←5;
 L[5]←9;L[6]←2;L[7]←6;L[8]←5;L[9]←3;
 PRINL(0,N);
 SORT(0,N,N);
 WRITE($0D);
 PRINL(0,N);
END.
```

ME,FORM,μPASCAL),勝手に設定したもの(PALL, FORTH), 示してないもの(LISP) と, バラバラになっていて統一性がありません。しかし, LISP以外の言語では, サブルーチンだけを示すのでは, メインルーチンとのかかわりが不明になってしまいます。実行させたときに何か結果が出るもの,

リスト10　FORMによるソート

```
0:       DIMENSION L(20)
1:       N=15
2:       DO 10 I=1,N
3:        L(I)=RND(9)+1
4:    10 CONTINUE
5:       CALL 3000
6:       CALL 1000
7:       WRITE(/)
8:       CALL 3000
9:       STOP
10: 1000 DO 20 I=1,(N-1)
11:        ID=N-I
12:        IX=L(ID)
13:        CALL 2000
14:    20 CONTINUE
15:       RETURN
16: 2000 K=ID+1
17:   100 IF(L(K)-IX) 200,200,
18:        IF(K-N) ,,200
19:        L(K-1)=L(K)
20:        K=K+1
21:       GOTO 100
22:   200 L(K-1)=IX
23:       RETURN
24: 3000 DO 30 M=1,N
25:        WRITE(L(M).I2)
26:    30 CONTINUE
27:       RETURN
28:       END
```

リスト11　μPASCALによるソート

```
ARRAY L[20];
VAR I,N;
PROC INST (X,S,E);
 BEGIN
  IF (S<=E)
   THEN
    IF (X)=L[S])
      THEN
       L[S-1]:=X
      ELSE
       BEGIN
        L[S-1]:=L[S];
        INST (X,S+1,E);
       END
   ELSE
    L[S-1]:=X;
 END;
PROC SORT (S,I,E);
 BEGIN
  IF (S<I) THEN
   BEGIN
    INST (L[I-1],I,E);
    SORT (S,I-1,E);
   END;
 END;
PROC PRINL (S,E);
 VAR I;
 BEGIN
  FOR I:=S TO E DO
   WRITE (L[I]);
 END;
BEGIN
 N:=14;
 FOR I:=0 TO N DO
  L[I]:=RND(9);
 PRINL (0,N);
 SORT (0,N,N);
 WRITE (/);
 PRINL (0,N);
END.
```

ということを考えて作ったため, リストのようになってしまいました。LISPでソートを行なうには, ソートされる数の列をSORTという関数の引数とします。

リストを比べてみれば, それぞれの言語のプログラムの構成がよくわかるのではないでしょうか。

おわりに

今回の特集で, MZではいろいろな言語が使えることがわかってもらえたと思います。まだこのほかにも, LSIアセンブラ, GAL, DRAW, FAST, CAP-Xなど興味深い言語が多く残っているのですが, ページの都合で割愛させていただきました。なお, この記事を書くための資料を集めてみて, その莫大な量に気づいたとき, どうすればよいか困惑してしまいましたが, なんとか今回の形にまとめることができました。もし, どれかの言語に少しでも興味を持ったら, その原典(参考文献に入れてある)にあたってみることをお勧めします。それぞれの言語のもっている魅力を, その製作者よりも十分に伝えることは不可能と思われますから。

リスト12　TL/1によるソート

```
1000 PROC INST,SORT,PRINL
1010 VAR I,N
1020 ARRAY L[20]
1030 BEGIN
1040  N:=14
1050  FOR I:=0 TO N DO
1060   L[I]:=RND(9)
1070  PRINL(0,N)
1080  SORT(0,N,N)
1090  WRITE(0:CRLF)
1100  PRINL(0,N)
1110 END
1120 INST(X,S,E)
1130  BEGIN
1140   IF(S>E)
1150    THEN
1160     L[S-1]:=X
1170    ELSE
1180      IF(X<L[S])
1190       THEN
1200        [ L[S-1]:=L[S]
1210          INST(X,S+1,E) ]
1220    ELSE
1230     L[S-1]:=X
1240  END
1250 SORT(S,I,E)
1260  BEGIN
1270   IF (S<I) THEN
1280    [ INST(L[I-1],I,E)
1290      SORT(S,I-1,E)    ]
1300  END
1310 PRINL(S,E)
1320  VAR I
1330  BEGIN
1340   FOR I:=S TO E DO
1350    WRITE(0:L[I],SPACE(1))
1360  END
```

リスト13　LISPによるソート

```
(I/O版) >(DEFINE INST (LAMBDA (X L) (COND ((NULL
         L) (CONS X L))((GREATERP X (CAR L)) (CO
        NS X L)) (T (CONS (CAR L) (INST X (CDR L
        ))))))))
        >(DEFINE SORT (LAMBDA (L) (COND ((NULL (
        CDR L)) L)(T (INST (CAR L) (SORT (CDR L)
        ))))))

(新版)  (DE SORT (L)
          (COND
            ((NULL (CDR L)) L)
            (T (INST (CAR L) (SORT (CDR L))))
          ))
        (DE INST (X L)
          (COND
            ((NULL L) (CONS X L))
            ((GREATERP X (CAR L)) (CONS X L))
            (T (CONS (CAR L)
                     (INST X (CDR L)) ))
          ))
```

リスト14　FORTH/Σによるソート

(エディタによるプログラム)

```
 1:        20 ARRAY L
 2:        : WOVER OVER OVER ;
 3:        : WROT ROT ROT ;
 4:        : INST
 5:            WOVER <= IF
 6:             WROT WOVER L A@ >= IF
 7:              1 - L A! DROP
 8:             ELSE
 9:              DUP L A@ OVER 1 - L A!
10:              1 + ROT INST
11:             THEN
12:            ELSE DROP 1 - L A! THEN
13:        ;
14:        : SORT
15:            WROT WOVER < IF
16:             1 - ROT OVER WOVER L A@
17:            ROT 1 + ROT
18:             INST
19:             SORT
20:            ELSE
21:             DROP DROP DROP
22:            THEN
23:        ;
24:        : PRINL
25:           DO I L A@ SPACE . LOOP
26:        ;
27:        3 0 L A! 1 1 L A! 4 2 L A!
28:        1 3 L A! 5 4 L A! 9 5 L A!
29:        2 6 L A! 6 7 L A! 5 8 L A!
30:        3 9 L A!
31:        CR
32:        10 0 PRINL
33:        0 9 9 SORT
34:        CR
35:        10 0 PRINL
```

リスト15　M-FORTHによるソート

(エディタによるプログラム)

```
FORGET L
$8000 CONST L
[ A! + !B ]
[ A@ + @B ]
[ SPACE " " S. ]
[ I1 DROP 1 - L A! ]
[ I2 OVER L A@ 3OVER ]
[ I3 OVER L A@ 2OVER 1 - L A! SWAP 1 + S
WAP ]
[ INST DDUP > IF I1 ELSE I2 > IF I3 INST
 ELSE I1 THEN THEN ]
[ S1 OVER DUP 1 - L A@ SWAP 2OVER INST S
WAP 1 - SWAP ]
[ S2 DROP DROP DROP ]
[ SORT OVER 3OVER > IF S1 SORT ELSE S2 T
HEN ]
[ PRINL DO I L A@ SPACE . LOOP ]
3 0 L A! 1 1 L A! 4 2 L A! 1 3 L A!
5 4 L A! 9 5 L A! 2 6 L A! 6 7 L A!
5 8 L A! 3 9 L A!
LF
10 0 PRINL
LF
0 9 9 SORT
LF
10 0 PRINL
```

インタプリタパスカルの拡張

Megumu Kudou
工藤 恵

はじめに

パスカルは，教育的でかつわかりやすいプログラムを書けるという特徴をもっていますが，シャープから供給されているインタプリタパスカルMZ-1Z004はテープ版で，ディスクに対するコマンドを含んでいません。

そこで，パスカルのコントロールコマンドを追加して，ディスクに対するセーブ，アペンドなどのコマンドを実行できるようにしてみました。

1 プログラムの入力方法

アセンブルリストもしくはダンプリストに従い，正しくプログラムを入力してください。また，ワークエリアはすべて00Hにしておきます。プログラムを入力し終わったら，実行の前に一旦，SAVEしておきましょう。

Sコマンドで，

S-adr. $62FB
E-adr. $745B
J-adr. $00B1

でテープにSAVEしてください。

次にモニタのJコマンドで

J-adr. $71F5

で実行させて，正常に動くかどうかを確認してください。正常に動作したら，3項のディスク版の作成に移ります。

ここでくれぐれも注意しなければならないのは，テープにSAVEする前にJ-71F5Hを実行すると，71F5H以降のメモリはパスカルのプログラムエリアとして使用され，入力したプログラムが保存されません。必ずJ-adr. $00B1でテープにSAVEしておきましょう。

2 プログラムの説明

パスカルは，プログラムエリアとして，62FBH以降を使用しているため，インタプリタを変更して，プログラムエリアを71F5H以降としました。また，コントロールコマンドは，V.1.0Aでは434FHで各コマンドへ分岐しているので，ディスクに対するコマンド追加をそこで変更しています。以上の処理は71F5H～7249Hで行なっています。

62FBH～6470Hは，ディスクに対するREADおよびWRITEのためのサブルーチンです。これは，MZ-2000のIPLリストおよびI/Oに掲載された飛岡氏のプログラムを参考にさせていただきました。

6CB0H～71F4Hはメインコマンドで，希望とするコントロールコマンドを実現しています。

724AH～745AHはCRTの下4行にファンクションキーを表示するためのプログラムで，スクロールエリアは0～20カラムです。前述のように，ここは1度実行されると，パスカルのプログラムエリアとして使用されます。

ディスクでのディレクトリエリアは，トラック01Hセクタ01Hからトラック01Hセクタ08Hまでで，FATエリアはトラック00Hセクタ02Hからトラック00Hセクタ05Hまでを使用しています。これはファイルの管理をしやすくするためで，したがって，シャープのDISK BASICなどのディスクとの互換性はまったくありません。しかし，使用するディスクは，DISK BASICのUtilityなどでイニシャライズしたものを使用してください。

また，処理速度向上のためにディレクトリエリアを1度にワークエリアに読み込むことにしたため，ワークエリアが大きくなりましたが，たとえばワークエリアを256バイトにすることも，もちろん可能です。しかしその場合，ディレクトリの読み方とFATの記入方式の変更が必要になります。

3 ディスク版の作成

以上で，パスカルにおいてもディスクに対するアクセスが可能になるわけですが，ここで，DISK BASICのように，IPLから直接BOOTする方法を示します。

1. 本プログラムをJ-adr. $00B1としてテープにSAVEしておきます。
2. DISK BASICを起動します。
3. UtilityをRUNさせ，リセットボタンを押してモニタに戻ります。
4. 何らかのユーティリティプログラムで，4000H～EFFFHを00Hにしておきます。
5. モニタの02B9H，02BAH，02BBHを21H，00H，70Hに変更します。
6. モニタのLコマンドでシャープのインタプリタパスカルMZ-

1Z004をテープからロードします。

7. モニタの02BA$_H$, 02BB$_H$をFB$_H$, D2$_H$に変更します（モニタのロードコマンドのパッチ）。

8. 1.で作成したテープを，同じくモニタのLコマンドでロードします。これは自動的にD2FB$_H$以降にロードされます。

9. 705D$_H$, 70AF$_H$, 70B0$_H$を58$_H$, F5$_H$, 71$_H$に変更します。

10. 4000$_H$以降に，01$_H$, 49$_H$, 50$_H$, 4C$_H$, 50$_H$, 52$_H$, 4F$_H$, 4D$_H$, 5A$_H$, 2D$_H$, 31$_H$, 5A$_H$, 30$_H$, 30$_H$, 34$_H$, 0D$_H$と書き込みます。

11. 4014$_H$, 4015$_H$に5A$_H$, 74$_H$と書き込みます。

12. 401E$_H$, 401F$_H$に30$_H$, 00$_H$と書き込みます。

13. 4100$_H$～417F$_H$に01$_H$を書き込みます。

14. 5000$_H$に80$_H$を書き込みます。

15. 145E$_H$, 145F$_H$, 1460$_H$, 1482$_H$, 1483$_H$, 1484$_H$, 14A3$_H$, 14A4$_H$, 14A5$_H$を00$_H$に変更します。

16. 144F$_H$, 1450$_H$を5A$_H$, A4$_H$とします。

17. ドライブ2にイニシャライズ済みのディスクを入れます。

18. J-adr. $12A0を実行してUtilityに戻り，Cコマンドで，From 1 to 2でマスターディスクを作成できます。

めんどうなようですが，基本的には4000$_H$～5000$_H$まではマスターの情報，7000$_H$～E45A$_H$をディスクの03$_H$トラック01$_H$セクタから書き込めば，マスターが作成可能なわけです。したがって，これを応用してゲームや他のシステムソフトなどを直接ディスクからIPLで起動できます。ただし，これはプログラムエリアが最大7FFF$_H$バイトまでのプログラムで，それ以上の大きなプログラム，たとえばテープ版のLOGOなどでは，まずローダーを起動させ，続いて本体をロードするといった手順をふむ必要があります。

また，ここで作成したディスクはシャープでいう，サブ・マスターディスクですが，大部分の方がご承知のように，Utilityの1465$_H$の20$_H$を18$_H$に変更すると，サブ・マスターディスクのコピーが可能です。

4 使用方法

パスカルの起動後，Q／コマンドでモニタに戻り，Lコマンドで実行させます。

このプログラムにより追加されたコントロールコマンドは，

Files (BASICのDIR), Append, Save, Delete, Lock, Unlock, Rename

です。すべてのコマンドは，前述のコマンドの頭文字＋／（スラッシュ）の形で使用します。すなわち，F／とかA／のように使ってください。以下，各コマンドを詳解します。

＊Files（ディスクのディレクトリの表示）

書式：F／（ドライブナンバー）

解説：入力はF／，F／1およびF／2のみが許されます。F／は現在のデフォルトドライブのディレクトリを表示します。また，デフォルトドライブを変更してファイルのディレクトリを表示させたい場合は，F／2のように入力するとドライブ2に入っているディスクのディレクトリを表示し，デフォルトドライブはドライブ2になります。

＊Append

書式：A／（ドライブナンバー）(ファイルネーム)

解説：使用方法は以下の3通りです。

①デフォルトドライブを変更しない場合。

たとえばA／と入力すると，Filename？と表示されます。以下ファイルネームを入力すると，そのファイルをアペンドします。

②デフォルトドライブを変更する場合。

現在デフォルトドライブがドライブ1で，ドライブ2からファイルをアペンドしたい場合，A／2と入力すると，Filename？と表示されます。1と同様にファイルネームを入力すると，ドライブ2から指定されたファイルをアペンドし，かつデフォルトドライブはドライブ2に変更されます。

③省略形

上記1，2は，A／ファイルネーム，A／1ファイルネーム，A／2ファイルネームのような省略形が許されており，ディレクトリ表示後すぐにアペンドする際には便利です。

＊Save, Delete, Lock, Unlock

書式：Appendと同様

解説：Appendと同様

＊Rename

書式：R／（ドライブナンバー）(旧ファイルネーム，新ファイルネーム)

解説：ほぼAppendと同様ですが，必ず，(コンマ) をはさんで2つのファイルネームを入力してください。コンマがなかったり，ファイルネームが1つの場合は実行されません。

おわりに

一応目につく限りのバグは除いたつもりですが，不十分な点もあるかと思います。読者の方のご意見をお待ちしています。

リスト

```
** Oh!MZ Check sum System V1.1 **
ADR. +0 +1 +2 +3 +4 +5 +6 +7 :sum.
62FB 3A 7A 64 B7 28 04 06 02 :2.03
6303 18 02 06 0A 3A 78 64 F6 :2.36
630B B4 32 7A 64 D3 DC 21 19 :3.7D
6313 3C 2B 7C B5 20 FB 10 F6 :3.B9
631B 21 00 00 2B 7C B5 CA 52 :2.99
6323 64 DB D8 2F 07 38 F4 3A :3.B3
632B 71 64 FE 01 28 09 B7 C2 :3.7E
6333 08 64 11 94 A2 18 03 11 :1.DF
-----------------------------1718
633B B4 A3 7B 21 BD 63 77 7A :4.04
```

```
6343 21 CC 63 77 3E 10 18 07 :2.34
634B 3A 7B 64 3D CA 08 64 32 :2.BE
6353 7B 64 3E D8 CD 33 64 2A :3.83
635B 76 64 22 7D 64 D9 0E DB :3.9F
6363 ED 5B 74 64 7B B7 28 01 :3.7B
636B 14 2A 72 64 D9 3E C4 CD :3.BC
6373 22 64 01 DB 06 DB D8 0F :3.2A
-------------------------------1A79
637B 38 CE 0F 38 F8 DB DB 05 :4.00
6383 20 F3 3E D8 CD 33 64 DB :4.68
638B DA D3 D9 2F 57 3A 7D 64 :4.27
6393 CB 3F BA 28 0A 2F D3 DB :3.D3
639B 3E 12 CD 33 64 18 CE 3A :2.D4
63A3 7D 64 CB 3F 2F D3 D9 38 :3.FE
63AB 03 AF 18 02 3E 01 5F 2F :1.99
63B3 D3 DD 3A 7E 64 2F D3 DA :4.A8
-------------------------------1D75
63BB D9 3E 94 CD 22 64 06 00 :3.04
63C3 DB D8 0F 38 22 0F 38 F8 :3.5B
63CB ED A2 20 F4 3A 7E 64 3C :3.FB
63D3 32 7E 64 FE 11 28 05 15 :2.65
63DB 20 E4 18 06 15 3E 01 32 :1.A8
63E3 7E 64 3E D8 CD 33 64 DB :4.37
63EB D8 2F B7 C2 4B 63 3A 7D :3.E5
63F3 64 3C 32 7D 64 7A B7 28 :3.0C
-------------------------------198F
63FB 08 D9 7B B7 C2 70 63 18 :3.C0
6403 9E 3D 32 7B 64 3A 79 64 :3.03
640B B7 28 07 3A 78 64 F6 80 :3.72
6413 18 01 AF 32 7A 64 D3 DC :3.87
641B DB D8 2F 32 7F 64 C9 E5 :4.A5
6423 F5 21 49 64 3E 30 77 F1 :3.99
642B CD 33 64 3E 38 77 E1 C9 :3.FB
6433 32 7C 64 2F D3 D8 C5 E5 :4.96
-------------------------------1E8B
643B 06 10 21 00 00 2B 7C B5 :1.93
6443 28 09 DB D8 2F 0F 38 F5 :3.4F
644B E1 C1 C9 10 ED 18 B6 11 :4.47
6453 62 64 CD B5 06 3E 00 32 :2.BE
645B 7A 64 D3 DC C3 01 13 44 :3.A8
6463 69 73 6B 20 6E 6F 74 20 :2.D8
646B 72 65 61 64 79 0D 26 6B :2.B3
6473 FE 45 20 04 3E B1 18 E2 :3.50
-------------------------------186A
647B E5 CD 04 57 E1 18 DB 3E :4.1F
6483 81 32 0C 6B 3E 01 32 0D :1.A8
648B 6B C3 32 4F 22 DD 21 34 :3.03
6493 5D 11 05 6B 06 00 0E 04 :0.F6
649B 7E 23 FE 27 28 04 FE 22 :3.12
64A3 20 31 DD 77 00 7E DD BE :3.BE
64AB 00 20 18 3A 1C 6B 3D 28 :1.5E
64B3 0A AF 21 11 6B B8 28 07 :2.3D
-------------------------------142B
64BB 3E B3 77 78 C3 36 5C 3E :3.73
64C3 B4 18 F7 FE 0D 28 0C 12 :3.14
64CB 23 13 04 0D 20 D7 11 54 :1.A3
64D3 14 18 D2 3E 53 CD 47 58 :2.FB
64DB 18 D1 E5 3E 01 32 13 6B :2.BD
64E3 3A 24 6B FE 50 3E 50 C4 :3.69
64EB 47 58 21 B4 13 CD 66 5B :3.15
64F3 E1 FE 3A 20 32 CD 27 5D :3.BC
-------------------------------181C
64FB 30 03 21 00 00 3A 1C 6B :1.15
6503 FE 01 28 2B 22 F9 6A 3E :3.15
650B A2 32 11 6B CD BB 12 06 :2.F0
6513 A2 6C 21 B4 13 11 A2 6C :3.15
651B 06 06 7E FE 3A 28 05 12 :2.01
6523 23 13 10 F6 C3 35 5C 3E :2.CE
652B 4E CD 47 58 C3 30 5C EB :3.F4
6533 2A AE 6C 2B 36 80 2B 72 :2.C2
-------------------------------15B4
653B 2B 73 18 E8 CD 27 5D 38 :3.27
6543 15 7C B5 08 CD F3 52 20 :3.80
654B 0D 08 32 1B 6B 18 04 AF :1.98
6553 32 1B 6B C3 30 5C 3E 55 :2.9A
655B CD 47 58 18 F6 3A 1C 6B :3.3B
6563 3D 20 37 E5 CD B0 5A D1 :4.21
656B FE 01 28 2E 08 CD 67 5E :2.EF
6573 08 B7 28 26 CD 88 4C DA :3.88
-------------------------------18AC
657B 86 4B E5 ED 5B 2A 6B 7E :4.11
6583 ED A0 FE 0D 20 F9 EB 22 :4.BE
658B 2A 6B CD BE 12 E1 3E 14 :3.65
6593 DA 8B 4B CD 96 5E 20 DC :4.6D
659B 18 22 CD B4 5E 3E B4 32 :3.3D
65A3 11 6B AF 32 03 6B CD 47 :2.DF
65AB 53 CD AF 12 DA 8B 4B CD :4.5E
65B3 72 4D CD 88 4C 38 C0 CD :4.25
-------------------------------1F40
65BB 96 5E 20 E1 C3 04 5C 2A :3.42
65C3 2D 6B 0E 06 1A 47 CD 57 :2.31
65CB 5B 3C 20 15 70 13 23 0D :1.7F
65D3 20 F2 ED 4B 2A 6B 70 23 :3.72
65DB 71 23 23 36 FF 22 2D 6B :2.A6
65E3 C9 AF 77 23 0D 20 FB 18 :3.52
65EB E9 11 F7 5D 18 03 11 04 :2.7E
65F3 5C CD F7 4C 28 12 CD 66 :3.D9
-------------------------------16B3
65FB 5B FE 3A CC 74 5B CD 72 :4.6D
6603 5A 3C 28 04 CD B5 12 C8 :3.1E
660B 3E 01 B7 C9 3E 2A 32 27 :2.B0
6613 6B C9 45 51 55 F0 82 5D :3.EE
661B 45 4E 54 F0 43 5C 44 45 :2.FF
6623 46 42 F0 B5 5C 44 45 46 :3.58
662B 57 F0 F2 5C 44 45 46 53 :3.B7
6633 F0 BF 5B 44 45 46 4D F0 :4.16
-------------------------------1C1D
663B 35 5D 45 58 54 F0 8C 5C :3.5B
6643 4D 41 43 52 4F F0 05 5E :2.C5
664B 45 4E 44 4D F0 04 5C 49 :2.BD
6653 46 F0 E4 5D 45 4E 44 49 :3.97
665B 46 F0 F7 5D 45 4E 44 F0 :4.51
6663 FD 5B 53 4B 50 F0 0F 5C :3.A1
666B 3B F0 30 5C 0D F0 30 5C :3.40
6673 4F 52 47 F0 BB 5B FF 4C :4.39
-------------------------------1BDF
667B 44 20 41 2C 28 42 43 29 :1.A7
6683 F1 0A 4C 44 20 41 2C 28 :2.40
668B 44 45 29 F1 1A 4C 44 20 :2.6D
6693 28 42 43 29 2C 41 F1 02 :2.36
669B 4C 44 20 28 44 45 29 2C :1.B6
66A3 41 F1 12 4C 44 20 41 2C :2.61
66AB 49 F2 ED 57 4C 44 20 41 :3.70
66B3 2C 52 F2 ED 5F 4C 44 20 :3.6C
-------------------------------137D
66BB 49 2C 41 F2 ED 47 4C 44 :3.6C
66C3 20 52 2C 41 F2 ED 4F 4C :3.59
66CB 44 20 53 50 2C 48 4C F1 :2.B8
66D3 F9 4C 44 20 53 50 2C 49 :2.C1
66DB 58 F2 DD F9 4C 44 20 53 :4.23
66E3 50 2C 49 59 F2 FD F9 50 :4.56
66EB 55 53 48 20 42 43 F1 C5 :3.4B
66F3 50 55 53 48 20 44 45 F1 :2.DA
-------------------------------1ADC
66FB D5 50 55 53 48 20 48 4C :2.C9
6703 F1 E5 50 55 53 48 20 41 :3.77
670B 46 F1 F5 50 55 53 48 20 :3.8C
6713 49 58 F2 DD E5 50 55 53 :4.4D
671B 48 20 49 59 F2 FD E5 50 :4.2E
6723 4F 50 20 42 43 F1 C1 50 :3.46
672B 4F 50 20 44 45 F1 D1 50 :3.5A
6733 4F 50 20 48 4C F1 E1 50 :3.75
-------------------------------1C5C
673B 4F 50 20 41 46 F1 F1 50 :3.78
6743 4F 50 20 49 58 F2 DD E1 :4.10
674B 50 4F 50 20 49 59 F2 FD :3.A0
6753 E1 45 58 20 44 45 2C 48 :2.9B
675B 4C F1 EB 45 58 20 41 46 :3.6C
6763 2C 41 46 27 F1 08 45 58 :2.70
676B 58 F1 D9 45 58 20 28 53 :3.5A
6773 50 29 2C 48 4C F1 E3 45 :3.52
-------------------------------1A4B
677B 58 20 28 53 50 29 2C 49 :1.E1
6783 58 F2 DD E3 45 58 20 28 :3.EF
678B 53 50 29 2C 49 59 F2 FD :3.89
6793 E3 4C 44 49 52 F2 ED B0 :4.9D
679B 4C 44 49 F2 ED A0 4C 44 :3.E8
67A3 44 52 F2 ED B8 4C 44 44 :4.01
67AB F2 ED A8 43 50 49 52 F2 :4.A7
```

```
67B3 ED B1 43 50 49 F2 ED A1 :4.FA
-------------------------------1F80
67BB 43 50 44 52 F2 ED B9 43 :4.04
67C3 50 44 F2 ED A9 44 41 41 :3.E2
67CB F1 27 43 50 4C F1 2F 4E :3.65
67D3 45 47 F2 ED 44 43 43 46 :3.7B
67DB F1 3F 53 43 46 F1 37 4E :3.82
67E3 4F 50 F1 00 48 41 4C 54 :2.B9
67EB F1 76 44 49 F1 F3 45 49 :4.66
67F3 F1 FB 49 4D 20 30 F2 ED :4.B1
-------------------------------1E18
67FB 46 49 4D 20 31 F2 ED 56 :3.62
6803 49 4D 20 32 F2 ED 5E 41 :3.66
680B 44 44 20 48 4C 2C 42 43 :1.ED
6813 F1 09 41 44 44 20 48 4C :2.77
681B 2C 44 45 F1 19 41 44 44 :2.88
6823 20 48 4C 2C 48 4C F1 29 :2.8E
682B 41 44 44 20 48 4C 2C 53 :1.FC
6833 50 F1 39 41 44 43 20 48 :2.AA
-------------------------------14E8
683B 4C 2C 42 43 F2 ED 4A 41 :3.67
6843 44 43 20 48 4C 2C 44 45 :1.F0
684B F2 ED 5A 41 44 43 20 48 :3.69
6853 4C 2C 48 4C F2 ED 6A 41 :3.96
685B 44 43 20 48 4C 2C 53 50 :2.0A
6863 F2 ED 7A 53 42 43 20 48 :3.99
686B 4C 2C 42 43 F2 ED 42 53 :3.71
6873 42 43 20 48 4C 2C 44 45 :1.EE
-------------------------------1758
687B F2 ED 52 53 42 43 20 48 :3.71
6883 4C 2C 48 4C F2 ED 62 53 :3.A0
688B 42 43 20 48 4C 2C 53 50 :2.08
6893 F2 ED 72 41 44 44 20 49 :3.83
689B 58 2C 42 43 F2 DD 09 41 :3.22
68A3 44 44 20 49 58 2C 44 45 :1.FE
68AB F2 DD 19 41 44 44 20 49 :3.1A
68B3 58 2C 49 58 F2 DD 29 41 :3.5E
-------------------------------1834
68BB 44 44 20 49 58 2C 53 50 :2.18
68C3 F2 DD 39 41 44 44 20 49 :3.3A
68CB 59 2C 42 43 F2 FD 09 41 :3.43
68D3 44 44 20 49 59 2C 44 45 :1.FF
68DB F2 FD 19 41 44 44 20 49 :3.3A
68E3 59 2C 49 59 F2 FD 29 41 :3.80
68EB 44 44 20 49 59 2C 53 50 :2.19
68F3 F2 FD 39 49 4E 43 20 42 :3.64
-------------------------------16CB
68FB 43 F1 03 49 4E 43 20 44 :2.75
6903 45 F1 13 49 4E 43 20 48 :2.8B
690B 4C F1 23 49 4E 43 20 53 :2.AD
6913 50 F1 33 49 4E 43 20 49 :2.B7
691B 58 F2 DD 23 49 4E 43 20 :3.44
6923 49 59 F2 FD 23 44 45 43 :3.80
692B 20 42 43 F1 0B 44 45 43 :2.6D
6933 20 44 45 F1 1B 44 45 43 :2.81
-------------------------------1616
693B 20 48 4C F1 2B 44 45 43 :2.9C
6943 20 53 50 F1 3B 44 45 43 :2.BB
694B 20 49 58 F2 DD 2B 44 45 :3.44
6953 43 20 49 59 F2 FD 2B 52 :3.71
695B 4C 43 41 F1 07 52 4C 41 :2.A7
6963 F1 17 52 52 43 41 F1 0F :3.30
696B 52 52 41 F1 1F 52 4C 44 :2.D7
6973 F2 ED 6F 52 52 44 F2 ED :5.15
-------------------------------19CF
697B 67 4A 50 20 28 48 4C 29 :2.06
6983 F1 E9 4A 50 20 28 49 58 :3.5D
698B 29 F2 DD E9 4A 50 20 28 :3.C3
6993 49 59 29 F2 FD E9 52 45 :4.3A
699B 54 20 4E 5A F1 C0 52 45 :3.64
69A3 54 20 5A F1 C8 52 45 54 :3.72
69AB 20 4E 43 F1 D0 52 45 54 :3.5D
69B3 20 43 F1 D8 52 45 54 20 :3.37
-------------------------------1ACA
69BB 50 4F F1 E0 52 45 54 20 :3.7B
69C3 50 45 F1 E8 52 45 54 20 :3.79
69CB 50 F1 F0 52 45 54 20 4D :3.89
69D3 F1 F8 52 45 54 49 F2 ED :4.FC
69DB 4D 52 45 54 4E F2 ED 45 :3.AA
69E3 52 45 54 F1 C9 49 4E 49 :3.85
69EB 52 F2 ED B2 49 4E 49 F2 :4.B5
69F3 ED A2 49 4E 44 52 F2 ED :4.9B
-------------------------------1FF8
69FB BA 49 4E 44 F2 ED AA 4F :4.6D
6A03 55 54 49 F2 ED A3 4F 54 :4.17
6A0B 49 52 F2 ED B3 4F 55 54 :4.25
6A13 44 F2 ED AB 4F 54 44 52 :4.07
6A1B F2 ED BB 4C 44 20 41 2C :3.B7
6A23 28 48 4C 29 F1 7E 4C 44 :2.E4
6A2B 20 42 2C 28 48 4C 29 F1 :2.64
6A33 46 4C 44 20 43 2C 28 48 :1.D5
-------------------------------1B84
6A3B 4C 29 F1 4E 4C 44 20 44 :2.A8
6A43 2C 28 48 4C 29 F1 56 4C :2.A4
6A4B 44 20 45 2C 28 48 4C 29 :1.BA
6A53 F1 5E 4C 44 20 48 2C 28 :2.9B
6A5B 48 4C 29 F1 66 4C 44 20 :2.C4
6A63 4C 2C 28 48 4C 29 F1 6E :2.BC
6A6B 4C 44 20 28 48 4C 29 2C :1.C1
6A73 41 F1 77 4C 44 20 28 48 :2.C9
-------------------------------13AB
6A7B 4C 29 2C 42 F1 70 4C 44 :2.D4
6A83 20 28 48 4C 29 2C 43 F1 :2.65
6A8B 71 4C 44 20 28 48 4C 29 :2.06
6A93 2C 44 F1 72 4C 44 20 28 :2.AB
6A9B 48 4C 29 2C 45 F1 73 4C :2.DE
6AA3 44 20 28 48 4C 29 2C 48 :1.BD
6AAB F1 74 4C 44 20 28 48 4C :2.D1
6AB3 29 2C 4C F1 75 41 44 44 :2.D0
-------------------------------1426
6ABB 20 41 2C 28 48 4C 29 F1 :2.63
6AC3 86 41 44 43 20 41 2C 28 :2.03
6ACB 48 4C 29 F1 8E 53 42 43 :3.14
6AD3 20 41 2C 28 48 4C 29 F1 :2.63
6ADB 9E 53 55 42 20 28 48 4C :2.64
6AE3 29 F1 96 41 4E 44 20 28 :2.CB
6AEB 48 4C 29 F1 A6 4F 52 20 :3.15
6AF3 28 48 4C 29 F1 B6 58 4F :3.33
-------------------------------1554
6AFB 52 20 28 48 4C 29 F1 AE :2.F6
6B03 43 50 20 28 48 4C 29 F1 :2.89
6B0B BE 49 4E 43 20 28 48 4C :2.74
6B13 29 F1 34 44 45 43 20 28 :2.62
6B1B 48 4C 29 F1 35 52 4C 43 :2.C4
6B23 20 28 48 4C 29 F2 CB 06 :2.C8
6B2B 52 4C 20 28 48 4C 29 F2 :2.95
6B33 CB 16 52 52 43 20 28 48 :2.58
-------------------------------14CE
6B3B 4C 29 F2 CB 0E 52 52 20 :3.04
6B43 28 48 4C 29 F2 CB 1E 53 :3.13
6B4B 4C 41 20 28 48 4C 29 F2 :2.84
6B53 CB 26 53 52 41 20 28 48 :2.67
6B5B 4C 29 F2 CB 2E 53 52 4C :3.51
6B63 20 28 48 4C 29 F2 CB 3E :3.00
6B6B FF 00 00 98 00 84 80 86 :3.21
6B73 80 C0 C0 C1 80 C0 08 80 :4.89
-------------------------------18FD
6B7B 08 C0 08 C0 08 C0 08 C0 :3.20
6B83 08 80 08 80 08 C8 00 C0 :2.A0
6B8B 80 C2 80 90 00 C0 80 88 :4.1A
6B93 00 C1 00 A8 00 A0 00 A3 :2.AC
6B9B 00 C0 20 C0 10 80 10 88 :2.C8
6BA3 00 80 80 82 80 80 20 00 :2.A2
6BAB 00 A0 00 C0 80 90 00 80 :2.F0
6BB3 80 D0 00 00 00 00 00 1C :1.6C
-------------------------------164C
6BBB 1B 18 1A 18 02 18 5A 00 :0.D9
6BC3 00 44 43 44 03 18 42 80 :1.A8
6BCB 01 40 05 81 01 41 07 42 :1.52
6BD3 00 43 01 44 02 45 03 48 :1.1A
6BDB 04 4C 05 4D 06 FF 2B 2D :1.FF
6BE3 20 2C 3A 3B 0D FF 4C 44 :2.5D
6BEB 20 23 2C 23 F1 40 01 4C :2.10
6BF3 44 20 23 2C 28 49 58 24 :1.A0
-------------------------------0CF9
6BFB 29 F3 DD 46 00 03 4C 44 :2.D2
6C03 20 23 2C 28 49 59 24 29 :1.86
6C0B F3 FD 46 00 03 4C 44 20 :2.E9
6C13 28 49 58 24 29 2C 23 F3 :2.58
6C1B DD 70 00 04 4C 44 20 28 :2.29
```

```
6C23 49 59 24 29 2C 23 F3 FD :3.2E
6C2B 70 00 04 4C 44 20 28 48 :1.94
6C33 4C 29 2C 24 F2 36 00 0F :1.FC
-------------------------------1280
6C3B 4C 44 20 28 49 58 24 29 :1.C6
6C43 2C 24 F4 DD 36 00 00 05 :2.5C
6C4B 4C 44 20 28 49 59 24 29 :1.C7
6C53 2C 24 F4 FD 36 00 00 05 :2.7C
6C5B 4C 44 20 41 2C 28 24 29 :1.92
6C63 F3 3A 00 00 06 4C 44 20 :1.E3
6C6B 23 2C 24 F2 06 00 02 4C :1.B9
6C73 44 20 28 24 29 2C 41 F3 :2.39
-------------------------------0FCC
6C7B 32 00 00 07 4C 44 20 48 :1.31
6C83 4C 2C 28 24 29 F3 2A 00 :2.0A
6C8B 00 0A 4C 44 20 42 43 2C :1.6B
6C93 28 24 29 F4 ED 4B 00 00 :2.A1
6C9B 0B 4C 44 20 44 45 2C 28 :1.98
6CA3 24 29 F4 ED 5B 00 00 0B :2.94
6CAB 4C 44 20 53 50 CD 6D 6D :2.FA
6CB3 CD E2 6C CD B1 6D 3E 01 :4.45
-------------------------------12B2
6CBB 32 A0 6C 21 80 64 22 A2 :3.07
6CC3 6C CD 27 05 28 0F CD F4 :3.5D
6CCB 6C CD 0D 6D CD 5B 6D FE :4.46
6CD3 41 28 02 18 EC 3E 00 32 :1.DF
6CDB 7A 64 D3 DC C3 01 13 21 :3.85
6CE3 00 08 22 74 64 21 80 64 :2.07
6CEB 22 72 64 3E 01 32 79 64 :2.46
6CF3 C9 2A A2 6C 11 80 6C 01 :2.FF
-------------------------------175A
6CFB 20 00 ED B0 C9 21 80 6C :3.93
6D03 ED 5B A2 6C 01 20 00 ED :3.64
6D0B B0 C9 3A 80 6C FE 80 28 :4.45
6D13 16 FE A0 C0 11 AD 70 CD :4.6F
6D1B B5 06 CD 43 6D 11 81 6C :3.36
6D23 CD B5 06 CD 64 07 C9 CD :4.56
6D2B 64 07 3A B1 6C FE 00 20 :2.B0
6D33 0A 11 A0 70 CD B5 06 CD :3.80
-------------------------------1D67
6D3B 64 07 C9 11 A7 70 18 F4 :3.68
6D43 3A 92 6C FE 00 20 0A CD :3.2D
6D4B 3A 06 CD 3A 06 CD 3A 06 :2.5A
6D53 C9 3E 2A CD 3C 06 18 F2 :3.4A
6D5B 2A A2 6C 11 20 00 19 22 :1.A4
6D63 A2 6C 3A A0 6C 3C 32 A0 :3.62
6D6B 6C C9 3E 01 32 76 64 32 :2.B2
6D73 77 64 C9 11 B1 70 CD B5 :4.58
-------------------------------1849
6D7B 06 3E 2F 32 E3 0B 11 00 :1.A4
6D83 11 CD E5 0B 11 0C 11 1A :2.16
6D8B FE 2C 28 07 FE 0D 28 0D :2.99
6D93 13 18 F4 3E 0D 12 13 1A :1.A9
6D9B FE 0D CA DF 6C 3A 00 11 :3.6B
6DA3 FE 0B CA DF 6C 3A 0C 11 :3.75
6DAB FE 0D CA DF 6C C9 3E 00 :4.27
6DB3 32 71 64 CD FB 62 C9 3E :4.38
-------------------------------173B
6DBB 01 32 71 64 CD FB 62 3A :3.6C
6DC3 7F 64 07 07 D0 CD 64 07 :2.F9
6DCB 11 37 71 CD B5 06 C3 D8 :3.DC
6DD3 6C CD 76 6D CD 6D 6D CD :4.90
6DDB 87 6D 3A A1 6C FE 05 20 :3.5E
6DE3 10 2A F7 62 01 01 00 ED :2.82
6DEB 42 01 F5 71 ED 42 DA DF :4.91
6DF3 6C CD E2 6C CD B1 6D 3E :4.B0
-------------------------------1DF2
6DFB 01 32 A0 6C 21 80 64 22 :2.66
6E03 A2 6C CD 27 05 CA D8 6C :4.15
6E0B CD F4 6C 3A 80 6C FE 00 :4.51
6E13 28 0D 11 0C 11 21 81 6C :1.71
6E1B 06 0F CD C8 05 28 18 CD :2.BC
6E23 5B 6D FE 41 20 DC 3A A1 :3.DE
6E2B 6C FE 05 CA 6B 6F 11 CE :3.F2
6E33 70 CD B5 06 C3 D8 6C 3A :4.39
-------------------------------1B02
6E3B A1 6C FE 02 20 04 3E 2A :2.99
6E43 18 06 FE 01 20 0E 3E 00 :1.89
6E4B 32 92 6C CD 00 6D CD BA :3.F1
6E53 6D C3 D8 6C 3A A1 6C FE :4.B9
6E5B 03 20 1D 3A 92 6C FE 2A :2.A0
6E63 20 09 11 BE 70 CD B5 06 :2.F0
6E6B C3 D8 6C 3E 00 32 80 6C :3.63
6E73 CD 00 6D CD BA 6D 18 14 :3.5A
-------------------------------1919
6E7B 3A A1 6C FE 06 28 40 FE :3.B1
6E83 04 28 71 11 DD 70 CD B5 :3.7D
6E8B 06 C3 D8 6C 11 20 71 CD :3.7C
6E93 B5 06 11 0C 11 CD B5 06 :2.71
6E9B 2A 94 6C 22 AC 6C CD 58 :3.89
6EA3 6F 2A 9C 6C 22 A6 6C CD :3.A2
6EAB 2E 6F 2A A6 6C 3A A5 6C :3.24
6EB3 06 00 70 23 3D FE 00 20 :1.F4
-------------------------------195E
6EBB F7 CD BA 6D C3 D8 6C 3A :5.2C
6EC3 92 6C FE 2A 28 9C 21 0C :3.17
6ECB 11 7E FE 0D 28 03 23 18 :2.00
6ED3 F8 23 11 81 6C 01 11 00 :2.2B
6EDB ED B0 3E 0D 32 91 6C 11 :3.28
6EE3 2B 71 CD B5 06 11 81 6C :3.22
6EEB CD B5 06 CD 00 6D CD BA :4.49
6EF3 6D C3 D8 6C 11 0C 71 CD :3.CF
-------------------------------1ADO
6EFB B5 06 11 81 6C CD B5 06 :3.41
6F03 2A 94 6C 22 74 64 3A 9E :2.FC
6F0B 6C 32 76 64 3A 9F 6C 32 :2.EF
6F13 77 64 2A F7 62 22 72 64 :3.56
6F1B ED 4B 94 6C 09 01 01 00 :2.43
6F23 ED 42 22 F7 62 CD B1 6D :4.95
6F2B C3 D8 6C 21 80 64 22 72 :3.A0
6F33 64 21 00 04 22 74 64 3E :1.C1
-------------------------------18BB
6F3B 00 32 76 64 3E 02 32 77 :1.F5
6F43 64 CD B1 6D C9 2A F7 62 :4.9B
6F4B 01 F5 71 ED 42 01 01 00 :2.98
6F53 09 22 AC 6C C9 2A AC 6C :3.4E
6F5B 3E 01 01 00 01 ED 42 38 :1.A8
6F63 03 3C 18 F6 32 A5 6C C9 :3.59
6F6B 11 16 71 CD B5 06 11 0C :2.3D
6F73 11 CD B5 06 CD 2E 6F CD :3.D0
-------------------------------1784
6F7B 48 6F CD 58 6F 21 7E 68 :3.52
6F83 22 AA 6C 21 A0 64 22 A8 :3.27
6F8B 6C 3E 00 32 A4 6C CD 7F :3.38
6F93 70 22 A8 6C 7E FE 01 20 :3.43
6F9B 08 23 3E 00 32 A4 6C 18 :1.C3
6FA3 ED 3A A4 6C 3C 32 A4 6C :3.B5
6FAB 3A A5 6C 47 3A A4 6C B8 :3.94
6FB3 28 06 23 CD 7F 70 18 D6 :2.FB
-------------------------------18FB
6FBB 3A A5 6C 4F 0D 06 00 ED :2.9A
6FC3 42 22 A6 6C 2A A6 6C 01 :2.B3
6FCB 80 04 ED 42 7C CB 27 CB :3.EC
6FD3 27 CB 27 CB 27 47 7D D6 :3.A5
6FDB 10 30 FC C6 11 32 AF 6C :3.60
6FE3 7D CB 3F CB 3F CB 3F CB :4.66
6FEB 3F 80 32 AE 6C 2A A6 6C :3.47
6FF3 3A A5 6C 06 01 70 23 3D :2.22
-------------------------------1A0D
6FFB FE 00 20 F7 CD BA 6D CD :4.D6
7003 6D 6D CD E2 6C CD B1 6D :4.E0
700B 3E 01 32 A0 6C 21 80 64 :2.82
7013 22 A2 6C CD F4 6C 3A 80 :4.17
701B 6C FE 00 28 09 CD 5B 6D :3.30
7023 FE 41 20 EF 18 6E 3E A0 :3.B2
702B 32 80 6C 3A AE 6C 32 9E :3.42
7033 6C 3A AF 6C 32 9F 6C 3E :3.3C
-------------------------------1DAF
703B 0D 32 1C 11 21 0C 11 11 :0.BB
7043 81 6C 01 11 00 ED B0 3E :2.DA
704B 00 32 92 6C 2A AC 6C 22 :2.94
7053 94 6C 2A A6 6C 22 9C 6C :3.66
705B CD 00 6D CD BA 6D 21 F5 :4.44
7063 71 22 72 64 2A AC 6C 22 :2.CD
706B 74 64 3A AE 6C 32 76 64 :3.38
7073 3A AF 6C 32 77 64 CD BA :3.E9
-------------------------------17C1
707B 6D C3 D8 6C E5 2A AA 6C :4.99
7083 ED 4B A8 6C ED 42 38 02 :3.B5
708B E1 C9 E1 11 EB 70 CD B5 :5.79
```

```
7093 06 C3 D8 6C 11 F9 70 CD :4.54
709B B5 06 C3 D8 6C 4D 41 53 :3.A3
70A3 54 45 52 0D 53 4C 41 56 :2.2E
70AB 45 0D 50 41 53 0D 46 69 :1.F2
70B3 6C 65 20 6E 61 6D 65 20 :2.B2
--------------------------------1C90
70BB 3F 20 0D 57 72 69 74 65 :2.77
70C3 20 70 72 6F 74 65 63 74 :3.21
70CB 65 64 0D 46 69 6C 65 20 :2.76
70D3 6E 6F 74 20 66 6F 75 6E :3.29
70DB 64 0D 41 6C 72 65 61 64 :2.BA
70E3 79 20 65 78 69 73 74 0D :2.D3
70EB 4E 6F 20 66 69 6C 65 20 :2.9D
70F3 73 70 61 63 65 0D 4E 6F :2.D6
--------------------------------1637
70FB 20 64 69 72 65 63 74 6F :3.0A
7103 72 79 20 73 70 61 63 65 :3.17
710B 0D 4C 6F 61 64 69 6E 67 :2.CB
7113 3A 20 0D 57 72 69 74 69 :2.76
711B 6E 67 3A 20 0D 44 65 6C :2.51
7123 65 74 69 6E 67 3A 20 0D :2.7E
712B 52 65 6E 61 6D 65 20 74 :2.EC
7133 6F 3A 20 0D 57 72 69 74 :2.7C
--------------------------------1599
713B 65 20 70 72 6F 74 65 63 :3.12
7143 74 65 64 20 64 69 73 6B :3.08
714B 0D CD 19 47 1B 1A FE 53 :2.C0
7153 20 04 3E 05 18 59 FE 41 :2.17
715B 20 04 3E 04 18 51 FE 44 :2.11
7163 20 04 3E 03 18 49 FE 46 :2.0A
716B 20 07 13 1A FE 2F 28 1B :1.C4
7173 E9 FE 4C 20 04 3E 02 18 :2.AF
--------------------------------137F
717B 36 FE 55 20 04 3E 01 18 :2.04
7183 2E FE 52 20 04 3E 06 18 :1.FE
718B 26 13 E9 13 1A FE 0D CA :3.24
7193 B0 6C FE 31 28 06 FE 32 :3.A9
719B 28 06 1B E9 3E 00 18 02 :1.8A
71A3 3E 01 32 78 64 13 1A FE :2.78
71AB 0D CA B0 6C 1B 1B E9 32 :3.44
71B3 A1 6C 13 1A FE 2F 28 01 :2.90
--------------------------------14A5
71BB E9 13 1A FE 0D CA D4 6D :4.2C
71C3 FE 31 28 23 FE 32 28 23 :2.F5
71CB FE 20 20 0B 13 1A FE 0D :2.81
71D3 CA D4 6D FE 20 28 F5 21 :4.67
71DB 0C 11 1A 77 23 13 1A FE :1.FC
71E3 0D 20 F8 77 C3 D7 6D 3E :3.E1
71EB 00 18 02 3E 01 32 78 64 :1.67
71F3 18 DA 3E C3 32 4C 43 21 :2.D5
--------------------------------1822
71FB 4C 71 22 4D 43 21 F5 71 :2.F6
7203 22 FA 12 22 57 13 22 74 :2.50
720B 15 22 B6 15 22 C0 15 22 :2.1B
7213 F5 16 22 25 17 22 9D 18 :2.40
721B 22 3E 58 3E 00 21 00 00 :1.17
7223 32 A0 12 22 A1 12 32 FD :2.E8
722B 12 22 FE 12 32 2B 1A 22 :1.DD
7233 2C 1A 22 2E 1A 22 30 1A :1.1C
--------------------------------1099
723B 22 32 1A 21 02 02 22 A9 :1.5E
7243 18 21 20 20 22 CC 18 CD :2.4C
724B 58 09 21 00 00 11 00 12 :0.A5
7253 01 BF 00 ED B0 3E 19 32 :2.B6
725B 0C 00 CD C3 06 06 15 3E :1.FB
7263 01 CD 3C 06 10 F9 11 BC :2.B6
726B 72 CD B5 06 11 DD 72 CD :4.27
7273 B5 06 11 2E 73 CD B5 06 :2.F5
--------------------------------12D2
727B 11 7F 73 CD B5 06 3E 14 :2.DD
```

```
7283 32 0C 00 CD C3 06 C3 A0 :3.37
728B 12 97 9B 9B 31 9B 9B 9B :3.E1
7293 9B 9B 9B 9B 32 9B 9B 9B :4.6F
729B 9B 9B 9B 9B 33 9B 9B 9B :4.70
72A3 9B 9B 9B 9B 34 9B 9B 9B :4.71
72AB 9B 9B 9B 9B 35 9B 9B 9B :4.72
72B3 9B 9B 9B 9B 36 9B 9B 9B :4.73
--------------------------------202A
72BB 9B 9B 9B 9B 37 9B 9B 9B :4.74
72C3 9B 9B 9B 9B 38 9B 9B 9B :4.75
72CB 9B 9B 9B 9B 39 9B 9B 9B :4.76
72D3 9B 9B 9B 9B 31 30 9B 9B :4.03
72DB 95 0D 9A 20 66 61 6C 73 :3.02
72E3 65 20 20 20 74 72 75 65 :2.85
72EB 20 20 20 64 6F 77 6E 74 :2.8C
72F3 6F 20 20 72 65 61 6C 3B :2.8E
--------------------------------1C03
72FB 20 20 20 63 68 61 72 3B :2.39
7303 20 20 20 66 6C 6F 61 74 :2.76
730B 28 20 20 74 72 75 6E 6B :2.9C
7313 28 20 20 20 75 6E 74 69 :2.48
731B 6C 20 20 20 74 68 65 6E :2.7B
7323 20 20 20 20 65 6C 73 65 :2.29
732B 20 9A 0D 9A 20 62 65 67 :2.AF
7333 69 6E 20 70 72 6F 63 65 :3.10
--------------------------------13F6
733B 64 75 72 65 20 66 75 6E :3.19
7343 63 74 69 6F 6E 20 3A 3D :2.B4
734B 20 20 69 6E 74 65 67 65 :2.BC
7353 72 3B 20 62 6F 6F 6C 65 :2.DE
735B 61 6E 3B 20 61 72 72 61 :2.D0
7363 79 5B 20 20 72 65 70 65 :2.C0
736B 61 74 20 20 20 77 68 69 :2.7D
7373 6C 65 20 20 20 20 65 6E :2.24
--------------------------------1598
737B 64 20 9A 0D 9B 9B 9B 9B :3.94
7383 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
738B 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
7393 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
739B 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
73A3 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
73AB 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
73B3 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
--------------------------------257C
73BB 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
73C3 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
73CB 9B 9B 9B 96 0D 62 65 67 :3.A2
73D3 69 6E 20 0D 70 72 6F 63 :2.B8
73DB 65 64 75 72 65 20 0D 66 :2.A8
73E3 75 6E 63 74 69 6F 6E 20 :3.20
73EB 0D 3A 3D 0D 69 6E 74 65 :2.41
73F3 67 65 72 3B 0D 62 6F 6F :2.C6
--------------------------------1AD9
73FB 6C 65 61 6E 3B 0D 61 72 :2.BB
7403 72 61 79 5B 0D 72 65 70 :2.FB
740B 65 61 74 20 0D 77 68 69 :2.AF
7413 6C 65 20 0D 65 6E 64 0D :2.42
741B 66 61 6C 73 65 0D 74 72 :2.FE
7423 75 65 0D 64 6F 77 6E 74 :3.13
742B 6F 20 0D 72 65 61 6C 3B :2.7B
7433 0D 63 68 61 72 3B 0D 66 :2.59
--------------------------------158C
743B 6C 6F 61 74 28 0D 74 72 :2.CB
7443 75 6E 63 28 0D 75 6E 74 :2.D2
744B 69 6C 20 0D 74 68 65 6E :2.B1
7453 20 0D 65 6C 73 65 20 0D :2.03
745B 00                      :0.00
------------------------------ -0A51
0006A48A
```

アセンブルリスト

```
   ##  Z80 ASSEMBLER 2Z-004C  <Expansion>  PAGE 01          12.29.83

01 0000                 ;-----------------------------------------------
02 0000                 ;
03 0000                 ;  Expansion of PASCAL interpriter  MZ-1Z004  V1.0
```

```
04 0000                 ;
05 0000                 ;  for mini-floppy disk operation  Ver. 1.1
06 0000                 ;
07 0000                 ;  by M. Kudoh              programmed: Nov. 15  '83
08 0000                 ;                           update    : Dec. 25  '83
```

```
0000                    ;
0000                    ;-----------------------------------------------------
0000                    ;
0000                    ;
0000                    ;   New commands calls Files ( F/ ), Append ( A/ ),
0000                    ;Save ( S/ ), Delete ( D/ ), Lock ( L/ ), Unlock
0000                    ;( U/ ) and Rename ( R/ ) have been developed for
0000                    ;mini-floppy disk operation.   This program should be
0000                    ;allocated to 62FBH ~, and sub-program "Lift-up &
0000                    ;Modification of interpreter" and "Function key
0000                    ;label" should be executed before boot up of PASCAL.
0000                    ;These sub-programs are deleted and the program areas
0000                    ;are used for PASCAL program areas when PASCAL is
0000                    ;booted.
0000                    ;
0000                    ;
0000                    ;┌──────────────────────────────────────┐
0000                    ;|                                      |
0000                    ;|          Disk control sub-routine    |
0000                    ;|                                      |
0000                    ;└──────────────────────────────────────┘
0000                    ;
0000                    ;disk control sub-routine for read and write
0000                    ;
0000                    ;work areas    :6471H ~ 647FH
0000                    ;program areas:62FBH ~ 6470H
0000                    ;
00DB  P                 CR:     EQU     DBH             ;status or command
00D9  P                 TR:     EQU     D9H             ;track :00H ~ 45H
00DA  P                 SR:     EQU     DAH             ;sector:01H ~ 10H
00DB  P                 DR:     EQU     DBH             ;data  :00H ~ FFH
00DC  P                 DM:     EQU     DCH             ;drive & motor
00DD  P                 HS:     EQU     DDH             ;head side select
0000                    ;
6471  P                 RWCMD:  EQU     6471H           ;read/write command
6472  P                 STAD:   EQU     RWCMD+01H       ;start address
6474  P                 SIZE:   EQU     RWCMD+03H       ;bytes size
6476  P                 TRCK:   EQU     RWCMD+05H       ;start track
6477  P                 SECT:   EQU     RWCMD+06H       ;start sector
6478  P                 DRNO:   EQU     RWCMD+07H       ;drive number
6479  P                 MTCMD:  EQU     RWCMD+08H       ;motor command
647A  P                 MTFLG:  EQU     RWCMD+09H       ;motor flag
647B  P                 ERFLG:  EQU     RWCMD+0AH       ;error flag
647C  P                 ERCMD:  EQU     RWCMD+0BH       ;error command
647D  P                 NTRCK:  EQU     RWCMD+0CH       ;next track
647E  P                 NSECT:  EQU     RWCMD+0DH       ;next sector
647F  P                 ERSTAT: EQU     RWCMD+0EH       ;error status
0000                    ;
62FB                            ORG     62FBH
62FB                    ;
62FB                    DSKCNT: ENT
62FB                    ;
```

```
   **  Z80 ASSEMBLER 2Z-004C  <Expansion>  PAGE 02          12.29.83

62FB                    ;----- Motor & Drive number set -----
62FB                    ;
62FB 3A7A64                     LD      A,(MTFLG)
62FE B7                         OR      A
62FF 2804                       JR      Z,Onesec
6301 0602                       LD      B,02H           ;0.2 sec delay
6303 1802                       JR      Less
6305 060A               Onesec: LD      B,0AH           ;1 sec delay
6307                    ;
6307 3A7B64             Less:   LD      A,(DRNO)
630A F684                       OR      84H
630C 327A64                     LD      (MTFLG),A
630F D3DC                       OUT     (DM),A          ;drive select
6311                    ;
6311 21193C             Waitlp: LD      HL,3C19H
6314 2B                 Inerlp: DEC     HL
6315 7C                         LD      A,H
6316 B5                         OR      L
6317 20FB                       JR      NZ,Inerlp
6319 10F6                       DJNZ    Waitlp
631B                    ;
631B                    ;----- Ready check -----
631B                    ;
631B 210000                     LD      HL,0000H
631E 2B                 Ready:  DEC     HL
631F 7C                         LD      A,H
6320 B5                         OR      L
6321 CA5264                     JP      Z,Nodisk        ;disk not ready
6324 DBD8                       IN      A,(CR)          ;status get
6326 2F                         CPL
6327 07                         RLCA
6328 38F4                       JR      C,Ready
632A                    ;
632A                    ;----- Read or Write command set -----
632A                    ;
632A 3A7164             Comset: LD      A,(RWCMD)
632D FE01                       CP      01H
632F 2809                       JR      Z,Wtset
6331 B7                         OR      A
6332 C20B64                     JP      NZ,Error
6335 1194A2                     LD      DE,A294H        ;read
6338 1803                       JR      Rdset
633A 11B4A3             Wtset:  LD      DE,A3B4H        ;write
633D 7B                 Rdset:  LD      A,E
633E 21BD63                     LD      HL,Comrw1+1
6341 77                         LD      (HL),A
6342 7A                         LD      A,D
6343 21CC63                     LD      HL,Comrw2+1
6346 77                         LD      (HL),A
6347                    ;
6347                    ;----- Program start -----
6347                    ;
6347 3E10                       LD      A,10H
6349 1807                       JR      Istart
634B                    ;
634B 3A7B64             Start:  LD      A,(ERFLG)
634E 3D                         DEC     A
634F CA0B64                     JP      Z,Error
6352 327B64             Istart: LD      (ERFLG),A
6355 3ED8                       LD      A,0D8H
```

```
   **  Z80 ASSEMBLER 2Z-004C  <Expansion>  PAGE 03          12.29.83

6357 CD3364                     CALL    Sub2
635A                    ;
635A 2A7664                     LD      HL,(TRCK)
635D 227D64                     LD      (NTRCK),HL
6360 D9                         EXX
6361 0EDB                       LD      C,DR
6363 ED5B7464                   LD      DE,(SIZE)
6367 7B                         LD      A,E
6368 B7                         OR      A
6369 2801                       JR      Z,Start1
```

```
636B 14                         INC     D
636C 2A7264             Start1: LD      HL,(STAD)
636F D9                         EXX
6370                    ;
6370                    ;----- Track sequence -----
6370                    ;
6370 3EC4               Trksq:  LD      A,C4H           ;read address
6372 CD2264                     CALL    Sub1
6375 01DB06                     LD      BC,06DBH
6378 DBD8               Trksq1: IN      A,(CR)
637A 0F                         RRCA
637B 38CE                       JR      C,Start
637D 0F                         RRCA
637E 38F8                       JR      C,Trksq1
6380 DBDB                       IN      A,(DR)
6382 05                         DEC     B
6383 20F3                       JR      NZ,Trksq1
6385 3ED8                       LD      A,0D8H          ;force interrupt
6387 CD3364                     CALL    Sub2
638A DBDA                       IN      A,(SR)
638C D3D9                       OUT     (TR),A
638E 2F                         CPL
638F 57                         LD      D,A
6390 3A7D64                     LD      A,(NTRCK)
6393 CB3F                       SRL     A
6395 BA                         CP      D
6396 280A                       JR      Z,Heads
6398 2F                         CPL
6399 D3DB                       OUT     (DR),A
639B 3E12                       LD      A,12H           ;seek
639D CD3364                     CALL    Sub2
63A0 18CE                       JR      Trksq
63A2                    ;
63A2 3A7D64             Heads:  LD      A,(NTRCK)
63A5 CB3F                       SRL     A
63A7 2F                         CPL
63A8 D3D9                       OUT     (TR),A
63AA 3803                       JR      C,Head1
63AC AF                         XOR     A
63AD 1802                       JR      Head2
63AF 3E01               Head1:  LD      A,01H
63B1 5F                 Head2:  LD      E,A
63B2 2F                         CPL
63B3 D3DD                       OUT     (HS),A
63B5 3A7E64                     LD      A,(NSECT)
63B8 2F                         CPL
63B9 D3DA                       OUT     (SR),A
63BB                    ;
63BB                    ;----- Sequencial read or write -----
63BB                    ;
```

```
   **  Z80 ASSEMBLER 2Z-004C  <Expansion>  PAGE 04          12.29.83

63BB D9                 Seqrw:  EXX
63BC 3E94               Comrw1: LD      A,94H           ;94H:seq. read
63BE CD2264                     CALL    Sub1            ;B4H:seq. write
63C1 0600               Seqrw1: LD      B,00H
63C3 DBD8               Seqrw2: IN      A,(CR)
63C5 0F                         RRCA
63C6 3822                       JR      C,Seqrw5
63C8 0F                         RRCA
63C9 38F8                       JR      C,Seqrw2
63CB EDA2               Comrw2: INI
63CD 20F4                       JR      NZ,Seqrw2
63CF 3A7E64                     LD      A,(NSECT)
63D2 3C                         INC     A
63D3 327E64                     LD      (NSECT),A
63D6 FE11                       CP      11H
63D8 2805                       JR      Z,Seqrw3
63DA 15                         DEC     D
63DB 20E4                       JR      NZ,Seqrw1
63DD 1806                       JR      Seqrw4
63DF 15                 Seqrw3: DEC     D
63E0 3E01                       LD      A,01H
63E2 327E64                     LD      (NSECT),A
63E5 3ED8               Seqrw4: LD      A,0D8H          ;force interrupt
63E7 CD3364                     CALL    Sub2
63EA DBD8               Seqrw5: IN      A,(CR)
63EC 2F                         CPL
63ED B7                         OR      A
63EE C24B63                     JP      NZ,Start        ;try again
63F1 3A7D64                     LD      A,(NTRCK)
63F4 3C                         INC     A
63F5 327D64                     LD      (NTRCK),A
63F8 7A                         LD      A,D
63F9 B7                         OR      A
63FA 2808                       JR      Z,End
63FC                    ;
63FC D9                         EXX
63FD 7B                         LD      A,E
63FE B7                         OR      A
63FF C27063                     JP      NZ,Trksq
6402 189E                       JR      Heads
6404                    ;
6404                    ;----- Out of sub-routine -----
6404                    ;
6404 3D                 End:    DEC     A
6405 327B64                     LD      (ERFLG),A
6408 3A7964             Error:  LD      A,(MTCMD)
640B B7                         OR      A
640C 2807                       JR      Z,End1
640E 3A7B64                     LD      A,(DRNO)
6411 F680                       OR      80H
6413 1801                       JR      End2
6415 AF                 End1:   XOR     A               ;motor off
6416 327A64             End2:   LD      (MTFLG),A
6419 D3DC                       OUT     (DM),A
641B DBD8                       IN      A,(CR)
641D 2F                         CPL
641E 327F64                     LD      (ERSTAT),A
6421 C9                         RET
6422                    ;
6422                    ;----- Sub-routines -----
```

```
   **  Z80 ASSEMBLER 2Z-004C  <Expansion>  PAGE 05          12.29.83

6422                    ;
6422 E5                 Sub1:   PUSH    HL
6423 F5                         PUSH    AF
6424 214964                     LD      HL,Sub2c
6427 3E30                       LD      A,30H
6429 77                         LD      (HL),A
642A F1                         POP     AF
642B CD3364                     CALL    Sub2
642E 3E38                       LD      A,38H
6430 77                         LD      (HL),A
6431 E1                         POP     HL
6432 C9                         RET
```

```
6433                    ;
6433 327C64     Sub2:   LD      (ERCMD),A
6436 2F                 CPL
6437 D3DB               OUT     (CR),A
6439                    ;
6439 C5                 PUSH    BC
643A E5                 PUSH    HL
643B 0610               LD      B,10H
643D 210000     Sub2a:  LD      HL,0000H
6440 2B         Sub2b:  DEC     HL
6441 7C                 LD      A,H
6442 B5                 OR      L
6443 2809               JR      Z,Sub2d
6445 DBDB               IN      A,(CR)
6447 2F                 CPL
6448 0F                 RRCA
6449 38F5       Sub2c:  JR      C,Sub2b
644B E1                 POP     HL
644C C1                 POP     BC
644D C9                 RET
644E 10ED       Sub2d:  DJNZ    Sub2a
6450 1BB6               JR      Error
6452                    ;
6452                    ;----- Display of "Disk not ready" -----
6452                    ;
6452 116264     Nodisk: LD      DE,Msgdsk
6455 CDB506             CALL    MSG
6458 3E00               LD      A,00H
645A 327A64             LD      (MTFLG),A       ;motor off
645D D3DC               OUT     (DM),A
645F C30113             JP      1301H           ;return to PASCAL
6462                    ;
6462 4469736B   Msgdsk: DEFM    "Disk not ready"
6466 206E6F74
646A 20726561
646E 6479
6470 0D                 DEFB    0DH
6471                    ;
6471                    ;┌──────────────────────┐
6471                    ;|                      |
6471                    ;|        Files+        |
6471                    ;|                      |
6471                    ;└──────────────────────┘
6471                    ;
6471                    ;main program
6471                    ;
6471                    ;Files, Append, Save, Delete, Lock, Unlock and
6471                    ;Rename commands are available.
```



```
6471                    ;
6471                    ;work areas    : 6480H ~ 6CAFH
6471                    ;program areas: 6CB0H ~ 7249H
6471                    ;initial start address: 71F5H (Modif)
6471                    ;
6480  P         Bufer1: EQU     6480H           ;directory & table area
6471                    ;
6CB0  P         Bufer2: EQU     Bufer1+800H     ;type of files
6CB1  P         Flname: EQU     Bufer2+01H      ;file name
6C92  P         Cklock: EQU     Bufer2+12H      ;lock & unlock check
6C94  P         Prgsiz: EQU     Bufer2+14H      ;program size
6C9C  P         Tablad: EQU     Bufer2+1CH      ;table address
6C9E  P         Dtrck:  EQU     Bufer2+1EH      ;track on disk
6C9F  P         Dsect:  EQU     Bufer2+1FH      ;sector on disk
6471                    ;
6CA0  P         Ckxfer: EQU     Bufer2+20H      ;number of xfer
6CA1  P         Ckcomd: EQU     Bufer2+21H      ;command check
6CA2  P         Xferad: EQU     Bufer2+22H      ;xfer address
6CA4  P         Cksect: EQU     Bufer2+24H      ;checked sector
6CA5  P         Ndsect: EQU     Bufer2+25H      ;required sector
6CA6  P         Wantad: EQU     Bufer2+26H      ;required address
6CA8  P         Chckad: EQU     Bufer2+28H      ;checked address
6CAA  P         Finlad: EQU     Bufer2+2AH      ;search final adr.
6CAC  P         Mprogs: EQU     Bufer2+2CH      ;program size
6CAE  P         Mtrck:  EQU     Bufer2+2EH      ;track on memory
6CAF  P         Msect:  EQU     Bufer2+2FH      ;sector on memory
6471                    ;
62F7  P         Prgend: EQU     62F7H           ;prog. end address
687E  P         Sekend: EQU     687EH           ;seek end address
64A0  P         Prtect: EQU     64A0H           ;seek first address
04B0  P         Offset: EQU     04B0H           ;offset for calculation
6471                    ;
6471                    ;MONITOR sub-routines
6471                    ;
0527  P         BRKEY:  EQU     0527H
0764  P         LETNL:  EQU     0764H
063A  P         PRNTS:  EQU     063AH
063C  P         PRNT:   EQU     063CH
06B5  P         MSG:    EQU     06B5H
1100  P         BUFER:  EQU     1100H
0BE3  P         KNUMBS: EQU     0BE3H
0BE5  P         GETL:   EQU     0BE5H
05C8  P         SAME:   EQU     05C8H
6471                    ;
6CB0                    ORG     6CB0H
6CB0                    ;
6CB0                    ;-----------------------
6CB0                    ;  FILES ( DIRECTORY )
6CB0                    ;-----------------------
6CB0                    ;
6CB0 CD6D6D     Files:  CALL    Rdfile
6CB3 CDE26C             CALL    Xfer1
6CB6 CDB16D             CALL    DSKRD
6CB9 3E01               LD      A,01H
6CBB 32A06C             LD      (Ckxfer),A
6CBE 21B064             LD      HL,Bufer1
6CC1 22A26C             LD      (Xferad),HL
6CC4                    ;
6CC4 CD2705     Floop:  CALL    BRKEY
6CC7 2B0F               JR      Z,Stpout
```



```
6CC9 CDF46C             CALL    Xfer2
6CCC CD0D6D             CALL    Fjudge
6CCF CD5B6D             CALL    Add20H
6CD2 FE41               CP      41H
6CD4 2802               JR      Z,Stpout
6CD6 1BEC               JR      Floop
6CD8                    ;
6CD8                    ;----- Moter stop -----
6CD8                    ;
6CD8 3E00       Stpout: LD      A,00H
6CDA 327A64             LD      (MTFLG),A
6CDD D3DC               OUT     (DM),A
6CDF                    ;
6CDF                    ;----- Out of command -----
```

```
6CDF                    ;
6CDF C30113     Allout: JP      1301H           ;return to PASCAL
6CE2                    ;
6CE2                    ;----- Xfer1, 800H bytes disk ↔ Bufer1 -----
6CE2                    ;
6CE2 21000B     Xfer1:  LD      HL,0B00H
6CE5 227464             LD      (SIZE),HL       ;size 0B00H
6CE8 218064             LD      HL,Bufer1
6CEB 227264             LD      (STAD),HL       ;bufer1 set
6CEE 3E01               LD      A,01H
6CF0 327964             LD      (MTCMD),A       ;motor on
6CF3 C9                 RET
6CF4                    ;
6CF4                    ;----- Xfer2, 20H bytes from Bufer1 to Bufer2 -----
6CF4                    ;
6CF4 2AA26C     Xfer2:  LD      HL,(Xferad)
6CF7 11B06C             LD      DE,Bufer2
6CFA 012000             LD      BC,0020H
6CFD EDB0               LDIR
6CFF C9                 RET
6D00                    ;
6D00                    ;----- Xfer3, reverse of Xfer2 -----
6D00                    ;
6D00 21B06C     Xfer3:  LD      HL,Bufer2
6D03 ED5BA26C           LD      DE,(Xferad)
6D07 012000             LD      BC,0020H
6D0A EDB0               LDIR
6D0C C9                 RET
6D0D                    ;
6D0D                    ;----- Judge of file name & others -----
6D0D                    ;
6D0D 3AB06C     Fjudge: LD      A,(Bufer2)
6D10 FE80               CP      80H
6D12 2B16               JR      Z,Master        ;master or slave
6D14 FEA0               CP      A0H
6D16 C0                 RET     NZ
6D17 11AD70             LD      DE,Msgpas
6D1A CDB506     Judlok: CALL    MSG
6D1D CD436D             CALL    Lockck
6D20 11B16C             LD      DE,Flname
6D23 CDB506             CALL    MSG
6D26 CD6407             CALL    LETNL
6D29 C9                 RET
6D2A                    ;
6D2A CD6407     Master: CALL    LETNL
6D2D 3AB16C             LD      A,(Flname)
```



```
6D30 FE00               CP      00H
6D32 200A               JR      NZ,Slave
6D34 11A070             LD      DE,Msgmst
6D37 CDB506     MSmsg:  CALL    MSG
6D3A CD6407             CALL    LETNL
6D3D C9                 RET
6D3E 11A770     Slave:  LD      DE,Msgslv
6D41 18F4               JR      MSmsg
6D43                    ;
6D43 3A926C     Lockck: LD      A,(Cklock)
6D46 FE00               CP      00H             ;unlock
6D48 200A               JR      NZ,Dsplok
6D4A CD3A06             CALL    PRNTS
6D4D CD3A06     Lokout: CALL    PRNTS
6D50 CD3A06             CALL    PRNTS
6D53 C9                 RET
6D54 3E2A       Dsplok: LD      A,2AH           ;lock
6D56 CD3C06             CALL    PRNT
6D59 18F2               JR      Lokout
6D5B                    ;
6D5B                    ;----- Addition of 20H -----
6D5B                    ;
6D5B 2AA26C     Add20H: LD      HL,(Xferad)
6D5E 112000             LD      DE,20H
6D61 19                 ADD     HL,DE
6D62 22A26C             LD      (Xferad),HL
6D65 3AA06C             LD      A,(Ckxfer)
6D68 3C                 INC     A
6D69 32A06C             LD      (Ckxfer),A
6D6C C9                 RET
6D6D                    ;
6D6D                    ;
6D6D                    ;----- Read files from disk to Bufer1 -----
6D6D                    ;
6D6D 3E01       Rdfile: LD      A,01H
6D6F 327664             LD      (TRCK),A        ;track 01H
6D72 327764             LD      (SECT),A        ;sector 01H
6D75 C9                 RET
6D76                    ;
6D76                    ;----- Display of "File name ?" -----
6D76                    ;
6D76 11B170     ?Fname: LD      DE,Msgfnm
6D79 CDB506             CALL    MSG
6D7C 3E2F               LD      A,2FH
6D7E 32E30B             LD      (KNUMBS),A
6D81 110011             LD      DE,BUFER
6D84 CDE50B             CALL    GETL
6D87                    ;
6D87 110C11     Renack: LD      DE,BUFER+0CH    ;check for Rename
6D8A 1A         Look:   LD      A,(DE)
6D8B FE2C               CP      2CH
6D8D 2807               JR      Z,Change
6D8F FE0D               CP      0DH
6D91 280D               JR      Z,Fnamok
6D93 13                 INC     DE
6D94 18F4               JR      Look
6D96                    ;
6D96 3E0D       Change: LD      A,0DH
6D98 12                 LD      (DE),A
6D99 13                 INC     DE
```



```
6D9A 1A                 LD      A,(DE)
6D9B FE0D               CP      0DH
6D9D CADF6C             JP      Z,Allout        ;omitt no file name
6DA0                    ;
6DA0 3A0011     Fnamok: LD      A,(BUFER)
6DA3 FE0B               CP      0BH
6DA5 CADF6C             JP      Z,Allout
6DA8 3A0C11             LD      A,(BUFER+0CH)   ;omitt no file name
6DAB FE0D               CP      0DH
6DAD CADF6C             JP      Z,Allout
6DB0 C9                 RET
6DB1                    ;
6DB1                    ;----- Read from disk -----
6DB1                    ;
6DB1 3E00       DSKRD:  LD      A,00H
6DB3 327164             LD      (RWCMD),A       ;read mode
```

```
17 6DB6 CDFB62                  CALL    DSKCNT
18 6DB9 C9                      RET
19 6DBA                 ;
20 6DBA                 ;----- Write to disk -----
21 6DBA                 ;
22 6DBA 3E01            DSKWT:  LD      A,01H
23 6DBC 327164                  LD      (RWCMD),A       ;write mode
24 6DBF CDFB62                  CALL    DSKCNT
25 6DC2 3A7F64                  LD      A,(ERSTAT)
26 6DC5 07                      RLCA
27 6DC6 07                      RLCA
28 6DC7 D0                      RET     NC
29 6DC8 CD6407                  CALL    LETNL
30 6DCB 113771                  LD      DE,Msgsof       ;write protected
31 6DCE CDB506                  CALL    MSG
32 6DD1 C3D86C                  JP      Stpout
33 6DD4                 ;
34 6DD4                 ;---------------------------------------
35 6DD4                 ;       Commands except for FILES
36 6DD4                 ;---------------------------------------
37 6DD4                 ;
38 6DD4 CD766D          Fname:  CALL    ?Fname
39 6DD7 CD6D6D          Comdin: CALL    Rdfile
40 6DDA CD876D                  CALL    Renack
41 6DDD                 ;
42 6DDD                 ;----- Check same file name -----
43 6DDD                 ;
44 6DDD 3AA16C                  LD      A,(Ckcomd)
45 6DE0 FE05                    CP      05H             ;SAVE
46 6DE2 2010                    JR      NZ,Carry
47 6DE4 2AF762                  LD      HL,(Prgend)     ;end address
48 6DE7 010100                  LD      BC,0001H
49 6DEA ED42                    SBC     HL,BC
50 6DEC 01F571                  LD      BC,Modif        ;start address
51 6DEF ED42                    SBC     HL,BC
52 6DF1 DADF6C                  JP      C,Allout        ;omitt no prog.size
53 6DF4                 ;
54 6DF4 CDE26C          Carry:  CALL    Xfer1
55 6DF7 CDB16D                  CALL    DSKRD
56 6DFA 3E01                    LD      A,01H
57 6DFC 32A06C                  LD      (Ckxfer),A
58 6DFF 21B064                  LD      HL,Bufer1
59 6E02 22A26C                  LD      (Xferad),HL
60 6E05 CD2705          Cloop:  CALL    BRKEY

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 10          12.29.83

01 6E08 CAD86C                  JP      Z,Stpout
02 6E0B CDF46C                  CALL    Xfer2
03 6E0E                 ;
04 6E0E 3AB06C                  LD      A,(Bufer2)
05 6E11 FE00                    CP      00H
06 6E13 2B0D                    JR      Z,Incres
07 6E15                 ;
08 6E15 110C11                  LD      DE,BUFER+0CH
09 6E18 21816C                  LD      HL,Flname
10 6E1B 060F                    LD      B,0FH
11 6E1D                 ;
12 6E1D CDCB05                  CALL    SAME
13 6E20 2B18                    JR      Z,Write
14 6E22                 ;
15 6E22 CD5B6D          Incres: CALL    Add20H
16 6E25 FE41                    CP      41H
17 6E27 20DC                    JR      NZ,Cloop
18 6E29                 ;
19 6E29 3AA16C                  LD      A,(Ckcomd)
20 6E2C FE05                    CP      05H
21 6E2E CA6B6F                  JP      Z,Save          ;jump to SAVE
22 6E31 11CE70          Dspnot: LD      DE,Msgnot
23 6E34 CDB506                  CALL    MSG
24 6E37 C3D86C                  JP      Stpout
25 6E3A                 ;
26 6E3A                 ;----- Carry out Lock & Unlock -----
27 6E3A                 ;
28 6E3A 3AA16C          Write:  LD      A,(Ckcomd)
29 6E3D FE02                    CP      02H
30 6E3F 2004                    JR      NZ,Unlock
31 6E41 3E2A                    LD      A,2AH
32 6E43 1806                    JR      Lock
33 6E45 FE01            Unlock: CP      01H
34 6E47 200E                    JR      NZ,DRAcom
35 6E49 3E00                    LD      A,00H
36 6E4B 32926C          Lock:   LD      (Cklock),A
37 6E4E CD006D                  CALL    Xfer3
38 6E51 CDBA6D                  CALL    DSKWT
39 6E54 C3D86C                  JP      Stpout
40 6E57                 ;
41 6E57                 ;----- Judge of Delete, Rename & Append ------
42 6E57                 ;
43 6E57 3AA16C          DRAcom: LD      A,(Ckcomd)
44 6E5A FE03                    CP      03H
45 6E5C 201D                    JR      NZ,RAcomd
46 6E5E 3A926C                  LD      A,(Cklock)
47 6E61 FE2A                    CP      2AH
48 6E63 2009                    JR      NZ,Delcon
49 6E65                 ;
50 6E65 11BE70          Protct: LD      DE,Msgwpt
51 6E68 CDB506                  CALL    MSG
52 6E6B C3D86C                  JP      Stpout
53 6E6E                 ;
54 6E6E 3E00            Delcon: LD      A,00H
55 6E70 32B06C                  LD      (Bufer2),A
56 6E73 CD006D                  CALL    Xfer3
57 6E76 CDBA6D                  CALL    DSKWT
58 6E79 1814                    JR      Delete          ;jump to DELETE
59 6E7B                 ;
60 6E7B                 ;----- Rename and Append Check -----

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 11          12.29.83

01 6E7B                 ;
02 6E7B 3AA16C          RAcomd: LD      A,(Ckcomd)
03 6E7E FE06                    CP      06H
04 6E80 2B40                    JR      Z,Rename        ;jump to RENAME
05 6E82 FE04                    CP      04H
06 6E84 2B71                    JR      Z,Append        ;jump to APPEND
07 6E86 11DD70                  LD      DE,Msgale
08 6E89 CDB506                  CALL    MSG
09 6E8C C3D86C                  JP      Stpout
10 6E8F                 ;
11 6E8F                 ;--------------------
12 6E8F                 ;       DELETE
13 6E8F                 ;--------------------
14 6E8F                 ;
15 6E8F 112071          Delete: LD      DE,Msgdel
16 6E92 CDB506                  CALL    MSG
17 6E95 110C11                  LD      DE,BUFER+0CH
18 6E9B CDB506                  CALL    MSG
19 6E9B 2A946C                  LD      HL,(Prgsiz)     ;program size
20 6E9E 22AC6C                  LD      (Mprogs),HL
21 6EA1 CD5B6F                  CALL    Csect
22 6EA4 2A9C6C                  LD      HL,(Tablad)     ;table address
23 6EA7 22A66C                  LD      (Wantad),HL
24 6EAA CD2E6F                  CALL    Rdtabl
25 6EAD 2AA66C                  LD      HL,(Wantad)
26 6EB0 3AA56C                  LD      A,(Ndsect)
27 6EB3 0600            WteOOH: LD      B,00H
28 6EB5 70                      LD      (HL),B
29 6EB6 23                      INC     HL
30 6EB7 3D                      DEC     A
31 6EB8 FE00                    CP      00H
32 6EBA 20F7                    JR      NZ,WteOOH
33 6EBC CDBA6D                  CALL    DSKWT
34 6EBF C3D86C                  JP      Stpout
35 6EC2                 ;
36 6EC2                 ;--------------------
37 6EC2                 ;       RENAME
38 6EC2                 ;--------------------
39 6EC2                 ;
40 6EC2 3A926C          Rename: LD      A,(Cklock)
41 6EC5 FE2A                    CP      2AH
42 6EC7 2B9C                    JR      Z,Protct
43 6EC9 210C11                  LD      HL,BUFER+0CH
44 6ECC 7E              Search: LD      A,(HL)
45 6ECD FE0D                    CP      0DH
46 6ECF 2B03                    JR      Z,Find
47 6ED1 23                      INC     HL
48 6ED2 18F8                    JR      Search
49 6ED4 23              Find:   INC     HL
50 6ED5 11B16C                  LD      DE,Flname
51 6ED8 011100                  LD      BC,11H
52 6EDB EDB0                    LDIR
53 6EDD 3E0D                    LD      A,0DH
54 6EDF 32916C                  LD      (Flname+10H),A  ;omitt long name
55 6EE2 112B71                  LD      DE,Msgren
56 6EE5 CDB506                  CALL    MSG
57 6EE8 11816C                  LD      DE,Flname
58 6EEB CDB506                  CALL    MSG
59 6EEE CD006D                  CALL    Xfer3
60 6EF1 CDBA6D                  CALL    DSKWT

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 12          12.29.83

01 6EF4 C3D86C                  JP      Stpout
02 6EF7                 ;
03 6EF7                 ;--------------------
04 6EF7                 ;       APPEND
05 6EF7                 ;--------------------
06 6EF7                 ;
07 6EF7 110C71          Append: LD      DE,Msglod
08 6EFA CDB506                  CALL    MSG
09 6EFD 11B16C                  LD      DE,Flname
10 6F00 CDB506                  CALL    MSG
11 6F03 2A946C                  LD      HL,(Prgsiz)     ;program size
12 6F06 227464                  LD      (SIZE),HL
13 6F09 3A9E6C                  LD      A,(Dtrck)       ;track in disk
14 6F0C 327664                  LD      (TRCK),A
15 6F0F 3A9F6C                  LD      A,(Dsect)       ;sector in disk
16 6F12 327764                  LD      (SECT),A
17 6F15 2AF762                  LD      HL,(Prgend)     ;start address
18 6F18 227264                  LD      (STAD),HL
19 6F1B ED4B946C                LD      BC,(Prgsiz)
20 6F1F 09                      ADD     HL,BC
21 6F20 010100                  LD      BC,0001H
22 6F23 ED42                    SBC     HL,BC
23 6F25 22F762                  LD      (Prgend),HL     ;end address
24 6F28 CDB16D                  CALL    DSKRD
25 6F2B C3D86C                  JP      Stpout
26 6F2E                 ;
27 6F2E                 ;--------------------
28 6F2E                 ;       SAVE
29 6F2E                 ;--------------------
30 6F2E                 ;
31 6F2E                 ;----- Read of byte table to Bufer1 -----
32 6F2E                 ;
33 6F2E 21B064          Rdtabl: LD      HL,Bufer1
34 6F31 227264                  LD      (STAD),HL       ;address Bufer1
35 6F34 210004                  LD      HL,0400H
36 6F37 227464                  LD      (SIZE),HL       ;size 0400H bytes
37 6F3A 3E00                    LD      A,00H
38 6F3C 327664                  LD      (TRCK),A        ;track 00H
39 6F3F 3E02                    LD      A,02H
40 6F41 327764                  LD      (SECT),A        ;sector 02H
41 6F44 CDB16D                  CALL    DSKRD
42 6F47 C9                      RET
43 6F48                 ;
44 6F48                 ;----- Calclation of program size -----
45 6F48                 ;
46 6F4B 2AF762          Cprogs: LD      HL,(Prgend)     ;end address
47 6F4B 01F571                  LD      BC,Modif        ;start address
48 6F4E ED42                    SBC     HL,BC
49 6F50 010100                  LD      BC,0001H
50 6F53 09                      ADD     HL,BC
51 6F54 22AC6C                  LD      (Mprogs),HL
52 6F57 C9                      RET
53 6F58                 ;
54 6F58                 ;----- Calculation of required sector -----
55 6F58                 ;
56 6F58 2AAC6C          Csect:  LD      HL,(Mprogs)     ;program size
57 6F5B 3E01                    LD      A,01H
58 6F5D 010001          Calsec: LD      BC,0100H
59 6F60 ED42                    SBC     HL,BC
60 6F62 3B03                    JR      C,Calend

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 13          12.29.83

01 6F64 3C                      INC     A
02 6F65 18F6                    JR      Calsec
03 6F67 32A56C          Calend: LD      (Ndsect),A
04 6F6A C9                      RET
05 6F6B                 ;
06 6F6B                 ;----- Seek free sect in table -----
07 6F6B                 ;
08 6F6B 111671          Save:   LD      DE,Msgsav
09 6F6E CDB506                  CALL    MSG
10 6F71 110C11                  LD      DE,BUFER+0CH
11 6F74 CDB506                  CALL    MSG
12 6F77 CD2E6F                  CALL    Rdtabl
13 6F7A CD486F                  CALL    Cprogs
14 6F7D CD5B6F                  CALL    Csect
15 6F80 217E68                  LD      HL,Sekend       ;final adr-01H for seek
16 6F83 22AA6C                  LD      (Finlad),HL
17 6F86 21A064                  LD      HL,Prtect       ;for table protection
18 6F89 22AB6C                  LD      (Chckad),HL
19 6F8C 3E00                    LD      A,00H
20 6F8E 32A46C                  LD      (Cksect),A
```

```
21 6F91 CD7F70          Save1:  CALL    Ckspc
22 6F94 22AB6C                  LD      (Chckad),HL
23 6F97 7E                      LD      A,(HL)
24 6F98 FE01                    CP      01H             ;occupied sector
25 6F9A 2008                    JR      NZ,Save2
26 6F9C 23                      INC     HL
27 6F9D 3E00                    LD      A,00H
28 6F9F 32A46C                  LD      (Cksect),A
29 6FA2 18ED                    JR      Save1
30 6FA4                 ;
31 6FA4 3AA46C          Save2:  LD      A,(Cksect)
32 6FA7 3C                      INC     A
33 6FA8 32A46C                  LD      (Cksect),A
34 6FAB 3AA56C                  LD      A,(Ndsect)
35 6FAE 47                      LD      B,A
36 6FAF 3AA46C                  LD      A,(Cksect)
37 6FB2 B8                      CP      B
38 6FB3 2806                    JR      Z,Save3
39 6FB5 23                      INC     HL
40 6FB6 CD7F70                  CALL    Ckspc
41 6FB9 18D6                    JR      Save1
42 6FBB                 ;
43 6FBB 3AA56C          Save3:  LD      A,(Ndsect)
44 6FBE 4F                      LD      C,A
45 6FBF 0D                      DEC     C
46 6FC0 0600                    LD      B,00H
47 6FC2 ED42                    SBC     HL,BC
48 6FC4 22A66C                  LD      (Wantad),HL
49 6FC7                 ;
50 6FC7                 ;----- Calc of TRCK & SECT from calc adress -----
51 6FC7                 ;
52 6FC7 2AA66C          Calc:   LD      HL,(Wantad)
53 6FCA 01B004                  LD      BC,Offset       ;offset values
54 6FCD ED42                    SBC     HL,BC
55 6FCF 7C                      LD      A,H
56 6FD0 CB27                    SLA     A
57 6FD2 CB27                    SLA     A
58 6FD4 CB27                    SLA     A
59 6FD6 CB27                    SLA     A
60 6FD8 47                      LD      B,A

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 14          12.29.83

01 6FD9 7D                      LD      A,L
02 6FDA D610            Calc1:  SUB     10H
03 6FDC 30FC                    JR      NC,Calc1
04 6FDE                 ;
05 6FDE C611                    ADD     A,11H
06 6FE0 32AF6C                  LD      (Msect),A
07 6FE3 7D                      LD      A,L
08 6FE4 CB3F                    SRL     A
09 6FE6 CB3F                    SRL     A
10 6FE8 CB3F                    SRL     A
11 6FEA CB3F                    SRL     A
12 6FEC 80                      ADD     A,B
13 6FED 32AE6C                  LD      (Mtrck),A
14 6FF0                 ;
15 6FF0                 ;----- Write of table to disk -----
16 6FF0                 ;
17 6FF0 2AA66C                  LD      HL,(Wantad)
18 6FF3 3AA56C                  LD      A,(Ndsect)
19 6FF6 0601            Inprnt: LD      B,01H           ;occupied
20 6FF8 70                      LD      (HL),B
21 6FF9 23                      INC     HL
22 6FFA 3D                      DEC     A
23 6FFB FE00                    CP      00H
24 6FFD 20F7                    JR      NZ,Inprnt
25 6FFF CDBA6D                  CALL    DSKWT
26 7002                 ;
27 7002                 ;----- Seek of free directory -----
28 7002                 ;
29 7002 CD6D6D                  CALL    Rdfile
30 7005 CDE26C                  CALL    Xfer1
31 7008 CDB16D                  CALL    DSKRD
32 700B 3E01                    LD      A,01H
33 700D 32A06C                  LD      (Ckxfer),A
34 7010 218064                  LD      HL,Bufer1
35 7013 22A26C                  LD      (Xferad),HL
36 7016 CDF46C          Cfree:  CALL    Xfer2
37 7019 3AB06C                  LD      A,(Bufer2)
38 701C FE00                    CP      00H
39 701E 2809                    JR      Z,Free
40 7020 CD5B6D                  CALL    Add20H
41 7023 FE41                    CP      41H
42 7025 20EF                    JR      NZ,Cfree
43 7027 186E                    JR      Notspc
44 7029                 ;
45 7029                 ;----- Write of file name to directc y -----
46 7029                 ;
47 7029 3EA0            Free:   LD      A,0A0H          ;PASCAL file
48 702B 32B06C                  LD      (Bufer2),A
49 702E 3AAE6C                  LD      A,(Mtrck)       ;track
50 7031 32E66C                  LD      (Dtrck),A
51 7034 3AAF6C                  LD      A,(Msect)       ;sec
52 7037 32F6C                   LD      (Dsect),A
53 703A 3E0D                    LD      A,0DH           ;omitt long name
54 703C 321C11                  LD      (BUFER+1CH),A
55 703F 210C11                  LD      HL,BUFER+0CH
56 7042 11816C                  LD      DE,Fname
57 7045 011100                  LD      BC,0011H
58 7048 EDB0                    LDIR
59 704A                 ;
60 704A 3E00                    LD      A,00H           ;unlock

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 15          12.29.83

01 704C 32926C                  LD      (Cklock),A
02 704F 2AAC6C                  LD      HL,(Mprogs)
03 7052 22946C                  LD      (Prgsiz),HL
04 7055 2AA66C                  LD      HL,(Wantad)
05 7058 229C6C                  LD      (Tablad),HL
06 705B CD006D                  CALL    Xfer3
07 705E CDBA6D                  CALL    DSKWT
08 7061                 ;
09 7061                 ;----- Write of program to disk -----
10 7061                 ;
11 7061 21F571                  LD      HL,Modif
12 7064 227264                  LD      (STAD),HL       ;start address
13 7067 2AAC6C                  LD      HL,(Mprogs)
14 706A 227464                  LD      (SIZE),HL       ;program size
15 706D 3AAE6C                  LD      A,(Mtrck)
16 7070 327664                  LD      (TRCK),A        ;write track
17 7073 3AAF6C                  LD      A,(Msect)
18 7076 327764                  LD      (SECT),A        ;write sector
19 7079 CDBA6D                  CALL    DSKWT
20 707C C3DB6C                  JP      Stpout
21 707F                 ;
22 707F                 ;----- Space check in directory -----
23 707F                 ;
24 707F E5              Ckspc:  PUSH    HL
25 7080 2AAA6C                  LD      HL,(Finlad)
26 7083 ED4BA86C                LD      BC,(Chckad)
27 7087 ED42                    SBC     HL,BC
28 7089 3802                    JR      C,Nofile
29 708B E1                      POP     HL
30 708C C9                      RET
31 708D                 ;
32 708D                 ;----- Display of "No file space" -----
33 708D                 ;
34 708D E1              Nofile: POP     HL
35 708E 11EB70                  LD      DE,Msgnfs
36 7091 CDB506                  CALL    MSG
37 7094 C3DB6C                  JP      Stpout
38 7097                 ;
39 7097                 ;----- Display of "No directory space" -----
40 7097                 ;
41 7097 11F970          Notspc: LD      DE,Msgnds
42 709A CDB506                  CALL    MSG
43 709D C3DB6C                  JP      Stpout
44 70A0                 ;
45 70A0                 ;--------------------
46 70A0                 ;       Messages
47 70A0                 ;--------------------
48 70A0                 ;
49 70A0 4D4153          Msgmst: DEFM    "MASTER"
50 70A4 4552
51 70A6 0D                      DEFB    0DH
52 70A7 534C4156        Msgslv: DEFM    "SLAVE"
53 70AB 45
54 70AC 0D                      DEFB    0DH
55 70AD 504153          Msgpas: DEFM    "PAS"
56 70B0 0D                      DEFB    0DH
57 70B1 46696C65        Msgfnm: DEFM    "File name ? "
58 70B5 206E616D
59 70B9 65203F20
60 70BD 0D                      DEFB    0DH

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 16          12.29.83

01 70BE 57726974        Msgwpt: DEFM    "Write protected"
02 70C2 65207072
03 70C6 6F746563
04 70CA 746564
05 70CD 0D                      DEFB    0DH
06 70CE 46696C65        Msgnot: DEFM    "File not found"
07 70D2 206E6F74
08 70D6 20666F75
09 70DA 6E64
10 70DC 0D                      DEFB    0DH
11 70DD 416C7265        Msgale: DEFM    "Already exist"
12 70E1 61647920
13 70E5 65786973
14 70E9 74
15 70EA 0D                      DEFB    0DH
16 70EB 4E6F2066        Msgnfs: DEFM    "No file space"
17 70EF 696C6520
18 70F3 73706163
19 70F7 65
20 70F8 0D                      DEFB    0DH
21 70F9 4E6F2064        Msgnds: DEFM    "No directory space"
22 70FD 69726563
23 7101 746F7279
24 7105 20737061
25 7109 6365
26 710B 0D                      DEFB    0DH
27 710C 4C6F6164        Msglod: DEFM    "Loading: "
28 7110 696E673A
29 7114 20
30 7115 0D                      DEFB    0DH
31 7116 57726974        Msgsav: DEFM    "Writing: "
32 711A 696E673A
33 711E 20
34 711F 0D                      DEFB    0DH
35 7120 44656C65        Msgdel: DEFM    "Deleting: "
36 7124 74696E67
37 7128 3A20
38 712A 0D                      DEFB    0DH
39 712B 52656E61        Msgren: DEFM    "Rename to: "
40 712F 6D652074
41 7133 6F3A20
42 7136 0D                      DEFB    0DH
43 7137 57726974        Msgsof: DEFM    "Write protected disk"
44 713B 65207072
45 713F 6F746563
46 7143 74656420
47 7147 6469736B
48 714B 0D                      DEFB    0DH
49 714C                 ;
50 714C                 ;
51 714C                 ;-----------------------
52 714C                 ;    New commands set
53 714C                 ;-----------------------
54 714C                 ;
55 714C CD1947          Upcomd: CALL    4719H           ;branch for addition
56 714F 1B                      DEC     DE
57 7150 1A                      LD      A,(DE)
58 7151 FE53            S/:     CP      53H             ;check S/
59 7153 2004                    JR      NZ,A/
60 7155 3E05                    LD      A,05H

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 17          12.29.83

01 7157 1859                    JR      Comdck
02 7159 FE41            A/:     CP      41H             ;check A/
03 715B 2004                    JR      NZ,D/
04 715D 3E04                    LD      A,04H
05 715F 1851                    JR      Comdck
06 7161 FE44            D/:     CP      44H             ;check D/
07 7163 2004                    JR      NZ,F/
08 7165 3E03                    LD      A,03H
09 7167 1849                    JR      Comdck
10 7169 FE46            F/:     CP      46H             ;check F/
11 716B 2007                    JR      NZ,L/
12 716D 13                      INC     DE
13 716E 1A                      LD      A,(DE)
14 716F FE2F                    CP      2FH
15 7171 2818                    JR      Z,?Drive
16 7173 E9                      JP      (HL)
17 7174 FE4C            L/:     CP      4CH             ;check L/
18 7176 2004                    JR      NZ,U/
19 7178 3E02                    LD      A,02H
20 717A 1836                    JR      Comdck
21 717C FE55            U/:     CP      55H             ;check U/
22 717E 2004                    JR      NZ,R/
23 7180 3E01                    LD      A,01H
24 7182 182E                    JR      Comdck
```

```
25 7184 FE52            R/:     CP      52H             ;check R/
26 7186 2004                    JR      NZ,Out
27 7188 3E06                    LD      A,06H
28 718A 1826                    JR      Comdck
29 718C 13              Out:    INC     DE
30 718D E9                      JP      (HL)
31 718E                 ;
32 718E 13              ?Drive: INC     DE
33 718F 1A                      LD      A,(DE)
34 7190 FE0D                    CP      0DH
35 7192 CAB06C                  JP      Z,Files
36 7195 FE31                    CP      31H
37 7197 2806                    JR      Z,Drive1
38 7199 FE32                    CP      32H
39 719B 2806                    JR      Z,Drive2
40 719D 1B                      DEC     DE
41 719E E9                      JP      (HL)
42 719F                 ;
43 719F 3E00            Drive1: LD      A,00H
44 71A1 1802                    JR      Drive?
45 71A3 3E01            Drive2: LD      A,01H
46 71A5 327B64          Drive?: LD      (DRND),A
47 71A8 13                      INC     DE
48 71A9 1A                      LD      A,(DE)
49 71AA FE0D                    CP      0DH
50 71AC CAB06C                  JP      Z,Files
51 71AF 1B                      DEC     DE
52 71B0 1B                      DEC     DE
53 71B1 E9                      JP      (HL)
54 71B2                 ;
55 71B2 32A16C          Comdck: LD      (Ckcomd),A
56 71B5 13                      INC     DE
57 71B6 1A                      LD      A,(DE)
58 71B7 FE2F                    CP      2FH
59 71B9 2801                    JR      Z,Comck
60 71BB E9                      JP      (HL)

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 18          12.29.83

01 71BC                 ;
02 71BC 13              Comck:  INC     DE
03 71BD 1A                      LD      A,(DE)
04 71BE FE0D                    CP      0DH
05 71C0 CAD46D                  JP      Z,Fname         ;input file name
06 71C3 FE31                    CP      31H
07 71C5 2823                    JR      Z,Setdr1        ;drive 1 set
08 71C7 FE32                    CP      32H
09 71C9 2823                    JR      Z,Setdr2        ;drivr 2 set
10 71CB FE20                    CP      20H
11 71CD 200B                    JR      NZ,Setfln
12 71CF                 ;
13 71CF 13              Comck1: INC     DE
14 71D0 1A                      LD      A,(DE)
15 71D1 FE0D                    CP      0DH
16 71D3 CAD46D                  JP      Z,Fname         ;input file name
17 71D6 FE20                    CP      20H
18 71D8 28F5                    JR      Z,Comck1        ;ignore
19 71DA                 ;
20 71DA 210C11          Setfln: LD      HL,BUFER+0CH
21 71DD 1A                      LD      A,(DE)
22 71DE                 ;
23 71DE 77              Comck2: LD      (HL),A          ;file name set
24 71DF 23                      INC     HL
25 71E0 13                      INC     DE
26 71E1 1A                      LD      A,(DE)
27 71E2 FE0D                    CP      0DH
28 71E4 20FB                    JR      NZ,Comck2
29 71E6 77                      LD      (HL),A
30 71E7 C3D76D                  JP      Comdin
31 71EA                 ;
32 71EA 3E00            Setdr1: LD      A,00H
33 71EC 1802                    JR      Setdrv
34 71EE 3E01            Setdr2: LD      A,01H
35 71F0 327B64          Setdrv: LD      (DRND),A        ;drive number set
36 71F3 18DA                    JR      Comck1
37 71F5                 ;
38 71F5                 ;
39 71F5                 ;-----------------------------------------
40 71F5                 ;  Lift-up & Modification of interpreter
41 71F5                 ;-----------------------------------------
42 71F5                 ;
43 71F5 3EC3            Modif:  LD      A,0C3H          ;for branch
44 71F7 324C43                  LD      (434CH),A
45 71FA 214C71                  LD      HL,Upcomd
46 71FD 224D43                  LD      (434DH),HL      ;for new commands set
47 7200                 ;
48 7200 21F571          Liftup: LD      HL,Modif
49 7203 22FA12                  LD      (12FAH),HL
50 7206 225713                  LD      (1357H),HL
51 7209 227415                  LD      (1574H),HL
52 720C 22B615                  LD      (15B6H),HL
53 720F 22C015                  LD      (15C0H),HL
54 7212 22F516                  LD      (16F5H),HL
55 7215 222517                  LD      (1725H),HL
56 7218 229D18                  LD      (189DH),HL
57 721B 223E58                  LD      (583EH),HL
58 721E                 ;
59 721E 3E00                    LD      A,00H
60 7220 210000                  LD      HL,0000H

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 19          12.29.83

01 7223 32A012                  LD      (12A0H),A
02 7226 22A112                  LD      (12A1H),HL
03 7229 32FD12                  LD      (12FDH),A
04 722C 22FE12                  LD      (12FEH),HL
05 722F 32B1A                   LD      (1A2BH),A
06 7232 222C1A                  LD      (1A2CH),HL
07 7235 222E1A                  LD      (1A2EH),HL
08 7238 22301A                  LD      (1A30H),HL
09 723B 22321A                  LD      (1A32H),HL
10 723E 210202                  LD      HL,0202H
11 7241 22A918                  LD      (18A9H),HL
12 7244 212020                  LD      HL,2020H
13 7247 22CC18                  LD      (18CCH),HL
14 724A                 ;
15 724A                 ;
16 724A                 ;┌──────────────────┐
17 724A                 ;│                  │
18 724A                 ;│  Function key label  │
19 724A                 ;│                  │
20 724A                 ;└──────────────────┘
21 724A                 ;
22 724A                 ;
23 724A                 ;Display of function key
24 724A                 ;
25 724A                 ;program area: 724AH ~ 745AH
26 724A                 ;scroll 0 ~ 20 column
27 724A                 ;
28 724A                 ;MONITOR sub-routines
29 724A                 ;
30 0958  P              CHR80:  EQU     0958H
31 06C3  P              SCRSET: EQU     06C3H
32 724A                 ;
33 724A CD5809                  CALL    CHR80
34 724D 21D073                  LD      HL,Klabel
35 7250 110012                  LD      DE,1200H
36 7253 018F00                  LD      BC,008FH
37 7256 EDB0                    LDIR
38 7258                 ;
39 7258 3E19            Kdisp:  LD      A,19H
40 725A 320C00                  LD      (000CH),A
41 725D CDC306                  CALL    SCRSET
42 7260                 ;
43 7260 0615                    LD      B,15H
44 7262 3E01                    LD      A,01H
45 7264 CD3C06                  CALL    PRNT
46 7267 10F9                    DJNZ    -05H
47 7269                 ;
48 7269 118C72                  LD      DE,Msgl&f
49 726C CDB506                  CALL    MSG
50 726F                 ;
51 726F 11DD72                  LD      DE,Msgky1
52 7272 CDB506                  CALL    MSG
53 7275                 ;
54 7275 112E73                  LD      DE,Msgky2
55 7278 CDB506                  CALL    MSG
56 727B                 ;
57 727B 117F73                  LD      DE,Msglin
58 727E CDB506                  CALL    MSG
59 7281                 ;
60 7281 3E14                    LD      A,14H

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 20          12.29.83

01 7283 320C00                  LD      (000CH),A
02 7286 CDC306                  CALL    SCRSET
03 7289                 ;
04 7289 C3A012                  JP      12A0H           ;jump to initialize
05 728C                 ;
06 728C 979B9B31        Msgl&f: DEFM    "┌──1──────2──────3──────4──────5"
07 7290 9B9B9B9B
08 7294 9B9B9B32
09 7298 9B9B9B9B
10 729C 9B9B9B33
11 72A0 9B9B9B9B
12 72A4 9B9B9B34
13 72A8 9B9B9B9B
14 72AC 9B9B9B35
15 72B0 9B9B9B9B                DEFM    "──────6──────7──────8"
16 72B4 9B9B9B36
17 72B8 9B9B9B9B
18 72BC 9B9B9B37
19 72C0 9B9B9B9B
20 72C4 9B9B9B38
21 72C8 9B9B9B9B                DEFM    "──────9──────10─┐"
22 72CC 9B9B9B39
23 72D0 9B9B9B9B
24 72D4 9B9B9B31
25 72D8 309B9B95
26 72DC 0D                      DEFB    0DH
27 72DD 9A206661        Msgky1: DEFM    "| false   true   downto  real|   "
28 72E1 6C736520
29 72E5 20207472
30 72E9 75652020
31 72ED 20646F77
32 72F1 6E746F20
33 72F5 20726561
34 72F9 6C3B2020
35 72FD 20
36 72FE 636B6172                DEFM    "char           |   float("
37 7302 3B202020
38 7306 666C6F61
39 730A 7428
40 730C 20207472                DEFM    "  trunk(   until   then    else |"
41 7310 756E6B28
42 7314 20202075
43 7318 6E74696C
44 731C 20202074
45 7320 68656E20
46 7324 20202065
47 7328 6C736520
48 732C 9A
49 732D 0D                      DEFB    0DH
50 732E 9A206265        Msgky2: DEFM    "| begin procedure function :=  "
51 7332 67696E20
52 7336 70726F63
53 733A 65647572
54 733E 65206675
55 7342 6E637469
56 7346 6F6E203A
57 734A 3D2020
58 734D 696E7465                DEFM    "integer        | boolean| array["
59 7351 67657233
60 7355 20626F6F

-

   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 21          12.29.83

01 7359 6C65616E
02 735D 3B206172
03 7361 7261795B
04 7365 20207265                DEFM    "  repeat   while    end |"
05 7369 70656174
06 736D 20202077
07 7371 68696C65
08 7375 20202020
09 7379 656E6420
10 737D 9A
11 737E 0D                      DEFB    0DH
12 737F 9B9B9B9B        Msglin: DEFM    "└──────────────────────────────"
13 7383 9B9B9B9B
14 7387 9B9B9B9B
15 738B 9B9B9B9B
16 738F 9B9B9B9B
17 7393 9B9B9B9B
18 7397 9B9B9B9B
19 739B 9B9B9B9B
20 739F 9B9B9B9B
21 73A3 9B9B9B9B                DEFM    "──────────────────────"
22 73A7 9B9B9B9B
23 73AB 9B9B9B9B
24 73AF 9B9B9B9B
25 73B3 9B9B9B9B
26 73B7 9B9B9B9B
27 73BB 9B9B9B9B                DEFM    "──────────────────┘"
28 73BF 9B9B9B9B
```

```
29 73C3 9B9B9B9B
30 73C7 9B9B9B9B
31 73CB 9B9B9B96
32 73CF 0D                      DEFB   0DH
33 73D0 62656769        Klabel1 DEFM   "begin "
34 73D4 6E20
35 73D6 0D                      DEFB   0DH
36 73D7 70726F63                DEFM   "procedure "
37 73DB 65647572
38 73DF 6520
39 73E1 0D                      DEFB   0DH
40 73E2 66756E63                DEFM   "function "
41 73E6 74696F6E
42 73EA 20
43 73EB 0D                      DEFB   0DH
44 73EC 3A3D                    DEFM   ":="
45 73EE 0D                      DEFB   0DH
46 73EF 696E7465                DEFM   "integer        ;"
47 73F3 6765723B
48 73F7 0D                      DEFB   0DH
49 73F8 626F6F6C                DEFM   "boolean        ;"
50 73FC 65616E3B
51 7400 0D                      DEFB   0DH
52 7401 61727261                DEFM   "array["
53 7405 795B
54 7407 0D                      DEFB   0DH
55 7408 72657065                DEFM   "repeat "
56 740C 617420
57 740F 0D                      DEFB   0DH
58 7410 7768696C                DEFM   "while "
59 7414 6520
60 7416 0D                      DEFB   0DH
```

```
   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 22          12.29.83

01 7417 656E64                  DEFM   "end"
02 741A 0D                      DEFB   0DH
03 741B 66616C73                DEFM   "false"
04 741F 65
05 7420 0D                      DEFB   0DH
06 7421 74727565                DEFM   "true"
07 7425 0D                      DEFB   0DH
08 7426 646F776E                DEFM   "downto "
09 742A 746F20
10 742D 0D                      DEFB   0DH
11 742E 7265616C                DEFM   "real           ;"
12 7432 3B
13 7433 0D                      DEFB   0DH
14 7434 63686172                DEFM   "char           ;"
15 7438 3B
16 7439 0D                      DEFB   0DH
17 743A 666C6F61                DEFM   "float("
18 743E 7428
19 7440 0D                      DEFB   0DH
20 7441 7472756E                DEFM   "trunc("
```

```
21 7445 6328
22 7447 0D                      DEFB   0DH
23 7448 756E7469                DEFM   "until "
24 744C 6C20
25 744E 0D                      DEFB   0DH
26 744F 7468656E                DEFM   "then "
27 7453 20
28 7454 0D                      DEFB   0DH
29 7455 656C7365                DEFM   "else "
30 7459 20
31 745A 0D                      DEFB   0DH
32 745B                 ;
33 745B                         END

_
```

```
   ** Z80 ASSEMBLER 2Z-004C <Expansion> PAGE 23          12.29.83

?Drive 718E  ?Fname 6D76  A/     7159  Add20H 6D5B  Allout 6CDF
Append 6EF7  BRKEY  0527  BUFER  1100  Bufer1 6480  Bufer2 6CB0
CHR80  0958  CR     00D8  Calc   6FC7  Calcl  6FDA  Calend 6F67
Calsec 6F5D  Carry  6DF4  Cfree  7016  Change 6D96  Chckad 6CA8
Ckcomd 6CA1  Cklock 6C92  Cksect 6CA4  Ckspc  707F  Ckxfer 6CA0
Cloop  6E05  Comck  71BC  Comck1 71CF  Comck2 71DE  Comdck 71B2
Comdin 6DD7  Comrw1 63BC  Comrw2 63CB  Comset 632A  Cprogs 6F48
Csect  6F58  D/     7161  DM     00DC  DR     00DB  DRAcom 6E57
DRNO   6478  DSKCNT 62FB  DSKRD  6DB1  DSKWT  6DBA  Delcon 6E6E
Delete 6EBF  Drive1 719F  Drive2 71A3  Drive? 71A5  Dsect  6C9F
Dsplok 6D54  Dspnot 6E31  Dtrck  6C9E  ERCMD  647C  ERFLG  647B
ERSTAT 647F  End    6404  End1   6415  End2   6416  Error  6408
F/     7169  Files  6CB0  Find   6ED4  Finlad 6CAA  Fjudge 6D0D
Fname  6C81  Floop  6CC4  Fname  6DD4  Fnamok 6DA0  Free   7029
GETL   0BE5  HS     00DD  Head1  63AF  Head2  63B1  Heads  63A2
Incres 6E22  Inerlp 6314  Inprnt 6FF6  Istart 6352  Judlok 6D1A
KNUMBS 0BE3  Kdisp  7258  Klabel 73D0  L/     7174  LETNL  0764
Less   6307  Liftup 7200  Lock   6E4B  Lockck 6D43  Lokout 6D4D
Look   6D8A  MSG    06B5  MSmsg  6D37  MTCMD  6479  MTFLG  647A
Master 6D2A  Modif  71F5  Mprogs 6CAC  Msect  6CAF  Msgale 70DD
Msgdel 7120  Msgdsk 6462  Msgfnm 70B1  Msgky1 72DD  Msgky2 732E
Msgl&f 72BC  Msglin 737F  Msglod 710C  Msgmst 70A0  Msgnds 70F9
Msgnfs 70EB  Msgnot 70CE  Msgpas 70AD  Msgren 712B  Msgsav 7116
Msgslv 70A7  Msgsof 7137  Msgwpt 70BE  Mtrck  6CAE  NSECT  647E
NTRCK  647D  Ndsect 6CA5  Nodisk 6452  Nofile 708B  Notspc 7097
Offset 04B0  Onesec 6305  Out    71BC  PRNT   063C  PRNTS  063A
Prgend 62F7  Prgsiz 6C94  Protct 6E65  Prtect 64A0  R/     7184
RAcomd 6E7B  RWCMD  6471  Rdfile 6D6D  Rdset  633D  Rdtabl 6F2F
Ready  631E  Renack 6D87  Rename 6EC2  S/     7151  SAME   05C4
SCRSET 06C3  SECT   6477  SIZE   6474  SR     00D8  STAD   6472
Save   6F6B  Save1  6F91  Save2  6FA4  Save3  6FBB  Search 6ECC
Sekend 687E  Seqrw  63BB  Seqrw1 63C1  Seqrw2 63C3  Seqrw3 63DF
Seqrw4 63E5  Seqrw5 63EA  Setdr1 71EA  Setdr2 71EE  Setdrv 71F0
Setfln 71DA  Slave  6D3E  Start  634B  Start1 634C  Stpout 6C1C
Sub1   6422  Sub2   6433  Sub2a  643D  Sub2b  6440  Sub2c  6449
Sub2d  644E  TR     00D9  TRCK   6476  Tablad 6C9C  Trksq  6370
Trksq1 6378  U/     717C  Unlock 6E45  Upcomd 714C  Waitlp 6311
Wantad 6CA6  Write  6E3A  Wte00H 6EB3  Wtset  633A  Xfer1  6CE2
Xfer2  6CF4  Xfer3  6D00  Xferad 6CA2
```


Oh!シリーズ掲載プログラムの リーダーズサービス

最寄りのパソコンショップで好評発売中！

Elder Manの「前戦基地」 X1用

テープ版　3,200円

本誌7月号で発表した，おもしろゲーム1等賞受賞作品「前戦基地」のカセットを，リーダースサービスの一環として始めました。

第2次大戦，ヨーロッパ戦線最大の攻防戦。連合軍は総力をあげてノルマンディ上陸作戦を敢行した。上陸後，ヨーロッパ大陸のドイツ軍を一掃すべく，前戦基地を築き，一斉攻撃の準備を整えている。ドイツ軍は死命を決するこの戦いに反撃を開始した。さあ，君がこの前戦基地を守りぬくのだ。

マッハ3「フライトシミュレータ」 MZ-5500用 PC-9801用

5インチフロッピーディスク　6,800円
8インチフロッピーディスク　7,800円
（MZ-5500用はマウス使用可、5インチのみ）

高速3Dグラフィックを駆使した，本格的フライトシミュレータ。完璧な3次元処理で，敵機，地形ともリアルに表現され，着陸時の振動や敵機の空中分解，失速もシミュレートされている。この迫力についていけるか!?

「スペース・エンパイア」 PC-9801用

5インチフロッピーディスク　5,800円
8インチフロッピーディスク　6,800円

君とコンピュータが，国連軍・GIGA軍に分かれて，銀河全域を背景に一大植民地争奪戦争を開始した。極めて過激で，帝国主義的・軍国主義的な，大型ウォー・シミュレーションゲーム。

お問い合せ先 (株)日本ソフトバンク出版部 〒102 東京都千代田区四番町2-1 TEL.03-261-4095

パソコン千夜一夜　第5夜

パソコン御三家と15ゲーム

FORESIGHT　峰岸　順二

✦✦✦✦✦✦✦✦✦✦まえがき✦✦✦✦✦✦✦✦✦✦

ワンボードマイコンを拡張してタイニィ（小さい）ベーシックを走らせていた53年ごろ，海の向うのアメリカではAPPLE II，PET，TRS-80という強力なマシンが発売されました。

これらは当時，パソコン御三家といわれ，マニア垂涎の的でしたが，今夜はまず，これらのマシンをふり返ってみましょう。

つぎに，そのころ創立されたキマグレコンピュータクラブ（きまこんクラブ）を紹介します。

同好者との楽しいおしゃべり，そしてQ＆Aによる技術アップのために数多くのマイコンクラブが生まれました。しかしながら，その大部分は数年も続かずに発展的に解消したようです。

きまこんクラブは相変わらずアクティブな活動を続けています。そして，パソコンにのめりこんだマニアがいかに大きく成長したかも，今夜の話を通じてわかると思います。

ゲームとしては，当時の会報から，会員創作の「15ゲーム」をとり上げ，MZに移植しました。すべてのシャープBASICに移植できます。30分もあればキーインできます。

それでは今夜も，MZにプログラムを入れてお楽しみください。PETが発売されたころをしのびながら。

✦✦✦✦✦あこがれのパソコン御三家✦✦✦✦✦

昭和53年ごろのマニアにとっての憧れはアメリカのパソコンでした。

TK-80BSなどのタイニーなものに比べ，非常に強力なベーシックを持ち，しかもスイッチONでBASICが走るのです。いろいろなゲームソフトの話題も海を越えて伝わってきました。

まだ，その頃のBSは，①スイッチを入れ，②TK-80ボード上にあるRESETキーを押してLED８個を０とし，③16進キーでBASICインタプリタのスタートアドレスE000をセットし，④RUNキーを押す。これで初めてBASICモードになるのでした。

表１はアメリカのいわゆる御三家，APPLE II，PET，TRSが発表されたころと日本のそれを比較したものです。またこの３種のマシンの性能を表２にまとめました。BSの仕様（注１）に比べていかに強力か，がわかると思います。

特にAPPLE IIは52年５月発売当時からカラーBASICであり，

表１　日米パソコン揺らん期年表

年	アメリカ	日　本
51	APPLE I (5)	TK-80 (9)
52	PET-2001 (4) APPLE II (5) TRS-80 1 (12)	第1回マイコンショウ (5) ソードM200 (9) TK-80BS (12)
53		ベーシックマスター (9) MZ-80K (12)

（　）は発表月

表２　アメリカ御三家の比較

	APPLE II	PET-2001	TRS-80
メーカー	アップル	コモドール	タンディRS
米国発売	52・5	52・4	52・12
価格	$1,298 (4K)	$595 (4K)	$399 (4K)
日本発売	52・10	53・6	53・3
価格	714,000 (16K)	298,000 (16K)	198,000 (4K)
CPU	6502	6502	Z-80
BASIC ROM	6K	8K	4K
色	カラー	白黒	白黒
ディスプレイ	280×192ドット	40×25行	32/64×16行
カセット	1500ボー	1000ボー	250ボー
主なゲームソフト	インベーダー スターウォーズ	ルナランダー スタートレック	潜水艦ゲーム チェス

そのハードはいまだに健在です。周辺機器も充実し，ソフトもウン万本といわれ，アップルマニアは強力な結束のもと，マイコンライフをたのしんでいます。

これに比べて日本のメーカーの激しい新機種の発表，毎年更新する旧マシンのブラッシュアップ。「互換性はある」とメーカーは言っても一部のBASICに限られ，マシン語になると疑問です。

この点，わがシャープはクリーン思想を堅持し，MZ-700でSP-5030が走るのは感激です。53年12月「マイコン博士」のキャッチフレーズでMZ-80K発表，195,000円でした。このころのプログラムテープを現在MZ-700で走らせていますが，これは他機種には絶対にできない芸当でしょう。

きまこんクラブ設立のころ

パソコンの組み立て，製作（ハード）も，プログラム作り（ソフト）も，その人の人生経験の範囲内を動き回っている。これはパソコンに限らず，すべてについて言えるでしょう。ルールや遊び方を知らないゲームのプログラムは作れない—ことを考えればわかると思います。

パソコンの世界を広げるために，マニアを呼び合うことについては昨夜もふれました。NECマイコンクラブニュースはB4版8ページ隔月発行でしたが紙面をユーザーに開放し，交流を求めるマニアの声の広場となりました。

これらの文通の結果，つぎつぎにクラブが誕生していきました。このニュース紙の54年7月，「きままな集りの場を」の呼びかけに応じて，私も安立隆一さんを中心としたきまこんクラブに入会させていただきました。

主力メンバーを2，3，紹介しましょう。

安立隆一氏　当時，足利工業大学電気工学科在学，クラブの創立者で会長ですが，結局パソコンのプロに。彼については後章で詳しくふれたいと思います。

佐々木徹氏　鯖江市武生高校の先生，現在マイコン各誌に成績処理などの学校での実務プログラムを発表している北陸の大御所です。福井マイコンクラブの代表者。彼ほどの人もビキナーのころはNECニュース紙に「レベルIIのPRINT文にはとても苦労しました。どうしても思うようにいかず，Bit Innを訪ねて質問したところPRINT文とCURSOR文を混ぜて使ったためとわかりました」と述べています(注2)。

片山博氏　名文をふるい，会報を編集しました。京都大学工学部化学科卒業，住友金属鉱山(株)中央研究所勤務，8080系のマシン語の使い手としては会のトップクラスです。

13才の楯一平君から73才の渡辺義太郎氏までユニークな会員が多く，多士済々です。

あるマニアの成長

この当時のマニアが成長して現在のパソコン界をリードし，あるいは縁の下の力持ちになっている，と第2夜にお話ししましたが，ここで前述の安立会長のお話をしてみましょう。

彼のパソコンへののめり込みはTK-80から始め，BSを経由しているのですが，彼が昭和54年，各誌に投稿したレターおよび自作プログラムのリストが表3です。

これを見るとソフト派と思われますが，彼の得意の分野はハードなのです。この後，大学4年生のとき，思いきってコンピュータのプロの世界へ飛び込み，ヒューモラボラトリへ入社，'56年に独立し，システム開発を本業としています。

パソコン界において，ジャパンドリームの若き実現者にならないかとひそかに期待しているのです。マイコン関係誌への寄稿もますます増加，著作を表4にまとめてみました。

Oh!MZの若き読者の皆さん。そして，今はビキナーであってもレベルはいくらでも高くなることができます。彼のように。

表3　あるマニアの昭和54年の活躍

No.	記事	文献
1	プログラム7点の交換を求める	N（No.4）　3月
2	ボールキャッチゲーム	I/O別—7　6月
3	きままな集りの場を　群馬県内BSのつどい	N（No.6）　7月
4	プログラム最近作8点の交換を求める	同上
5	オプションコマンドの自動書き込み	N（No.7）　9月
6	バイオリズムグラフプリントプログラム	同上
7	きまこんクラブ募集　会費月額60円	同上
8	プログラム交換を求む。近作8点	同上
9	宇宙のクズ屋ゲーム	RAM　10月
10	ゴキブリタイジゲーム	RAM　11月
11	ロボット対UFO	I/O別—9　12月
12	馬とび将棋	同上

N　NECクラブニュース
I/O別　I/O別冊 BASICゲーム徹底研究

表4　あるマニアの著作リスト

書名	発行
PC-6001　プログラミング入門	ナツメ社
パソコン工作入門	同上
PC-8001 mkII　プログラミング入門	同上
FM-7　グラフィックス入門	同上
MZ-700　ゲームライブラリ	同上
PC-8801　漢字ワードプロセッサー入門	同上
PC-6001 mkII　プログラミング入門	同上
面白マイコン	小学館
PC-8001 mkII　アニメーショングラフィック	ナツメ社
PC-8801 mkII　アニメーショングラフィック	同上

15ゲーム

きまこんクラブでは片山さんの編集した会報が魅力的でした。全国にまたがっている広い範囲の会員の寄稿，マイコンライフの

リレー報告，自作プログラムの発表など，彼の筆にかかると生き生きとしたものです。

今夜のゲームは，会員の中西洋一さんのオリジナルを安立，片山さんが議論し，片山さんが可能型と不可能型を研究して可能型だけを出題するように改良した「15ゲーム」です（注3）。リスト中のFOR~NEXTの段付：はSP-5030の場合は除いてください。

誰でもごぞんじのゲームです。プログラムをRUNさせると，CRT画面にゲーム盤が図1のように表示されます。これを，たとえば図2のように並べかえるものです。

最初の駒の配置によって，実は「不可能型」があるのです。最終配置が図3のようになるものは不可能型です。したがって15枚の配置をランダムに行なうと½の確率で不可能型が出ることになります。

この可能型か不可能型かを判定し，不可能型の場合は可能型におきかえて出題するようにしたのが片山さんの改良版で，これから紹介するものがそれです。両方の型の判定については，会報中で何回か議論されてにぎやかでした。

オリジナルプログラムは，配列が1元で1文字@（）だけのレベルI，改良版はレベルIIで書かれており，今夜は私が両者を初めてまとめてみました。

〔遊び方〕

駒の動かし方は，空地の行または列にある駒を指定してやれば1～3枚の札を同時に移動させることができます。3枚を一度に動かしても1手とカウントし，CRTに手数が表示されます。駒移動のやり方は中西さんが大いに工夫されたところです。

作者，中西洋一さんは宇治市の住人，電々公社にお勤めでコンピュータのデータ通信がお仕事。BSが発表される前，まずTK-80にRAMを増設し，TVインタフェイスを接続して月刊アスキーのTiny BASICを走らせたという数少ないエリートマニアです。53年2月，雑誌bitのエディタ・アセンブラを走らせ，4月にそのソーステープを作り，53年6月にはすでにミニフロッピーを購入，7月にそのDOSを完成させているという，当時の私達には神様みたいな方でした。

短いプログラムです。走らせてみてください。そしてアメリカ御三家が発表された「パソコン元年」のころのゲームを楽しんでください。このくらいの長さならば，解読も容易で，きっと参考になるアルゴリズムもあると思います。

✦✦✦✦✦✦✦✦あるゲームの一生✦✦✦✦✦✦✦✦

55年3月，再び東京本社に転勤，倉敷はわずか1年半の生活でした。

このころ，3名の高校2年生の情熱によってクラブFORESIGHTが誕生するのですが，ここ2夜，クラブ誕生の話が続いたので明日の夜はちょっと変わった話をいたしましょう。

あるゲームソフトがTK-80BSによって誕生し，パソコンのグレードアップによってブラッシュアップされてもてはやされ，一時はベストセラーにまで上昇，やがて改良版の出現によって消え去っていく，「あるゲームソフトの一生」のお話です。

ゲームとしては，当時のマニアの話題となった，宇宙の彼方，イスカンダルを舞台としたものを候補にしています。

（注1） パソコン千夜一夜　第4夜，Oh！MZ　59年4月号
（注2） NECマイコンクラブニュース　No.4，54年3月，8ページ
（注3） 片山博　続15ゲームの解析　きまこんクラブ会報No.13，92ページ（55年3月）

図1　ゲームスタート

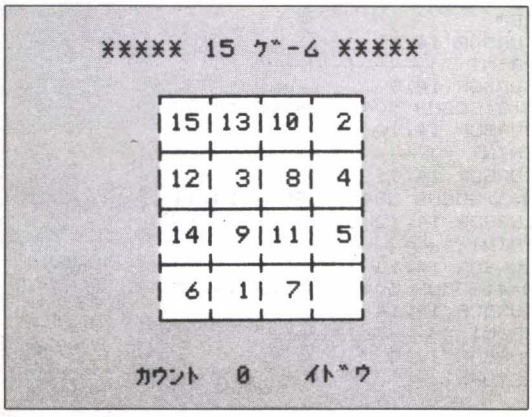


図2　最終ゲーム盤

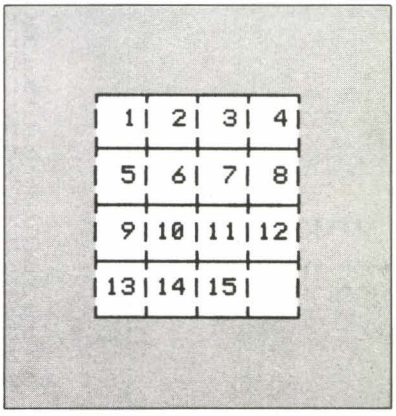
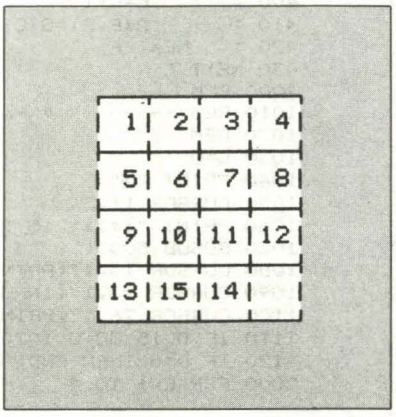


図3　不可能型



リスト

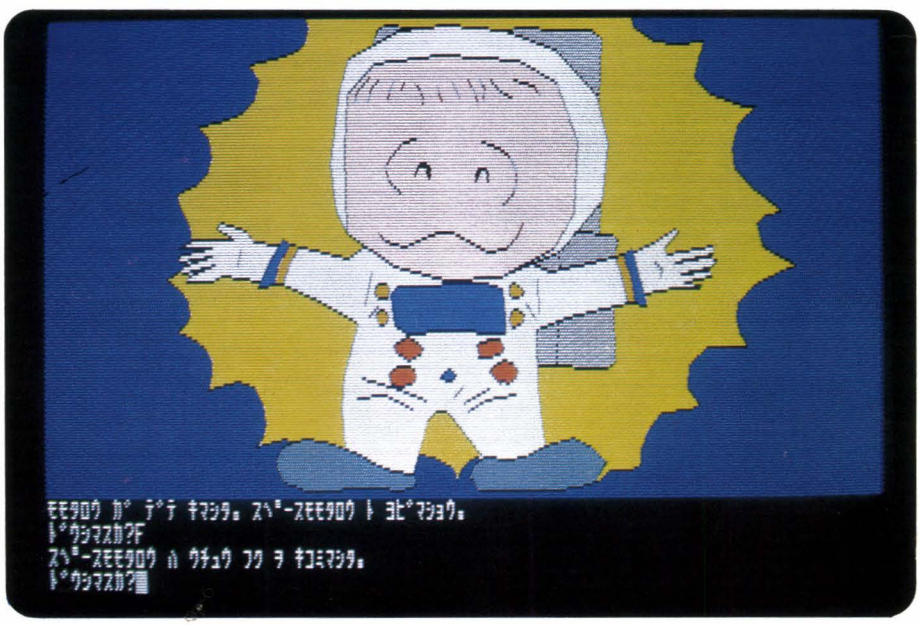
```
10 REM ********************************
15 REM
20 REM              15 ｹﾞｰﾑ
25 REM
30 REM       ﾏｼﾝ MZ-700(K/C),2000
35 REM
40 REM ORIGINAL  ﾅｶﾆｼ ﾖｳｲﾁ (BS L-1)
45 REM      ｷﾏｺﾝ ｸﾗﾌﾞ ｶｲﾎｳ NO.10 54/12
50 REM REVISION  ｶﾀﾔﾏ ﾋﾛｼ  (BS L-2)
55 REM      ｷﾏｺﾝ ｸﾗﾌﾞ ｶｲﾎｳ NO.13 55/3
60 REM ｲｼｮｸ       ﾐﾈｷﾞｼ ｼﾞｭﾝｼﾞ
65 REM      OH!MZ ﾊﾟｿｺﾝ ｾﾝﾔｲﾁｱ   59/5
70 REM      ｻｸｾｲ  59/2/26
75 REM
80 REM ********************************
100 REM
110 REM  --------  ｼｮｷ ｾｯﾃｲ  --------
120 REM
130 DIM A(5,5),G(16),T(16)
140 FOR J=1 TO 16
150 :   G(J)=J
160 NEXT J
170 FOR J=1 TO 31
180 :   K=INT(RND(1)*15+1)
190 :   L=INT(RND(1)*15+1)
200 :   G=G(K):G(K)=G(L):G(L)=G
210 NEXT J
220 FOR J=1 TO 15
230 :   T(J)=G(J)
240 NEXT J
250 C=0
260 D=0
270 FOR J=1 TO 14
280 :   IF T(J)<T(J+1) GOTO 310
290 :   T=T(J):T(J)=T(J+1):T(J+1)=T
300 :   D=1:C=C+1
310 NEXT J
320 IF D>0 THEN 260
330 IF C-2*INT(C/2)=0 GOTO 360
340 K=INT(RND(1)*14+1)
350 G=G(K):G(K)=G(K+1):G(K+1)=G
360 REM
370 C=0
380 FOR E=1 TO 4
390 :   FOR F=1 TO 4
400 :       C=C+1
410 :       A(E,F)=G(C)
420 :   NEXT F
430 NEXT E
1000 REM
1010 REM -------  ｹﾞｰﾑ ｶﾞﾒﾝ  ---------
1020 REM
1030 C=0
1040 PRINT "■"
1050 CURSOR 11,3
1060 PRINT "***** 15 ｹﾞｰﾑ *****"
1070 GOSUB 5000
1080 CURSOR 13,17:PRINT "ｶｳﾝﾄ ";C
1090 CURSOR 23,17:INPUT "ｲﾄﾞｳ ";D
1100 CURSOR 26,17:PRINT "       "
1110 IF D>15 GOTO 1070
1120 IF D=0 THEN END
2000 FOR E=1 TO 4
2010 :   FOR F=1 TO 4
2020 :       IF A(E,F)=D GOTO 2050
2030 :   NEXT F
2040 NEXT E
2050 FOR G=1 TO 4
2060 :   IF A(G,F)=16 GOTO 3000
2070 NEXT G
2080 FOR H=1 TO 4
2090 :   IF A(E,H)=16 GOTO 4040
2100 NEXT H
2110 GOTO 1070
3000 IF E>G GOTO 4000
3010 Q=E+1
3020 FOR J=G TO Q STEP -1
3030 :   A(J,F)=A(J-1,F)
3040 NEXT J
3050 A(E,F)=16
3060 C=C+1
3070 GOTO 1070
4000 FOR J=G TO E
4010 :   A(J,F)=A(J+1,F)
4020 NEXT J
4030 GOTO 3050
4040 IF F>H GOTO 4090
4050 FOR J=H TO F STEP -1
4060 :  A(E,J)=A(E,J-1)
4070 NEXT J
4080 GOTO 3050
4090 FOR J=H TO F
4100 :   A(E,J)=A(E,J+1)
4110 NEXT J
4120 GOTO 3050
5000 REM
5010 CURSOR 14,6:PRINT "┌─┬─┬─┬─┐"
5020 N=1:GOSUB 5040
5030 GOTO 5130
5040 CURSOR 14,5+2*N:PRINT "|";
5050 FOR B=1 TO 4
5060 :   IF A(N,B)=16 GOTO 5110
5070 :   IF A(N,B)<10 THEN PRINT " ";
5080 :   N$=STR$(A(N,B)):PRINT N$;"|";
5090 NEXT B
5100 RETURN
5110 PRINT "  |";
5120 GOTO 5090
5130 REM
5140 CURSOR 14,8
5150 PRINT "├─┼─┼─┼─┤"
5160 CURSOR 14,9
5170 N=2:GOSUB 5040:CURSOR 14,9
5180 CURSOR 14,10
5190 PRINT "├─┼─┼─┼─┤"
5200 CURSOR 14,11
5210 N=3:GOSUB 5040:CURSOR 14,11
5220 CURSOR 14,12
5230 PRINT "├─┼─┼─┼─┤"
5240 CURSOR 14,13
5250 N=4:GOSUB 5040:CURSOR 14,13
5260 CURSOR 14,14
5270 PRINT "└─┴─┴─┴─┘"
5280 CURSOR 1,13
5290 RETURN
```

パッケージソフト紹介マニュアル------74
これがプロのテクニックだ------90

殺し屋の罠に気をつけて

デゼニランドは，ハドソンから発表された初のアドベンチャーゲーム。ハドソンは，今年はハンパなソフトはださないっ！　と，なかなかの鼻息だそうですが，このデゼニランドはその意気込みを示しているかな？　X1用のメディアは3インチCDとテープの両方がありますが，ここはやはり5インチFD版も発売してほしいところですね。編集部では3インチCD版でテストランさせました。

プログラムはIPLスタートで，起動にBASICが不要な点は評価できます。もし起動時に，以前にセーブしたゲームの途中結果があれば，自動的にそのシーンから始まりますが，最初はもちろん，そんなことはなく，タイトルが現われます。アドベンチャーゲームの場合問題になるのはペイントの速度ですが，BASICのPAINTよりは速いルーチンを使っているようで，それほど待たされるわけではありません(もっと速いことに越したことはないが)。最初に「START」と打ち込むとゲームが始まります。そこは遊園地デゼニランドの門ですが，チケットがないと入れない。さて，どうするか…という具合にゲームが始まります。このゲームの目的は，黄金で作られているという臼，「三月磨臼」を見つけ出すことです。アドベンチャーゲームとしては，ごく普通の謎ですね。なんとかデゼニランドの外をうろつき回って，アドベンチャーゲームの鉄則「なんでも手あたりしだいにひっかきまわせ」と「なんでも取れる物はとれ」を駆使して，いよいよデゼニランドの内に入ります。入った所は広場。先に進むと6つのお店が並んで買い物ができますが，予算には限度があるので，すべての品を買うわけにはいきません。そこでじっくりと品を選ばなければなりません。ここでの買い物のしかたによっては，のちのちひどいことになるのは目にみえています。さて，ここで一応買い物



をすませてから，後は，それぞれの見せ物を回りながら謎を解いていくのですが，相当に意地の悪い謎もあります。

最初に目の前に現れる見せ物は，「瀬戸内海の海賊」です。中に入ると例によって，壁に絵がかかっていたり，封印の貼ってあるキャビネットがあったりします。よーく考えて，取れる物は全部取りましょう。ほんとに全部ですよ。

次にこの見せ物の中を見て回ることになるわけですが，殺しの罠に気をつけてください。それと，LOOKコマンドを忘れずに。ここでも，取れる物は，なんでも取って，使いましょう。

ここを出ると，次にやってくるのが「ジャングル・クロース」。まあようするにボートに乗ってジャングル探検ですな。ここでは，最初に何を買ったかが問題になってきますよ。

というぐあいに，進んでいくわけですが，もはやこの先は秘密ですので，みなさんで謎を解いてください。難易度はかなりのものですよ。

このデゼニランドは，多少欠点もありますが，速度や，100以上の画面数からして，日本のレベルから考えれば出色の一作というところでしょう。　(高野庸一)

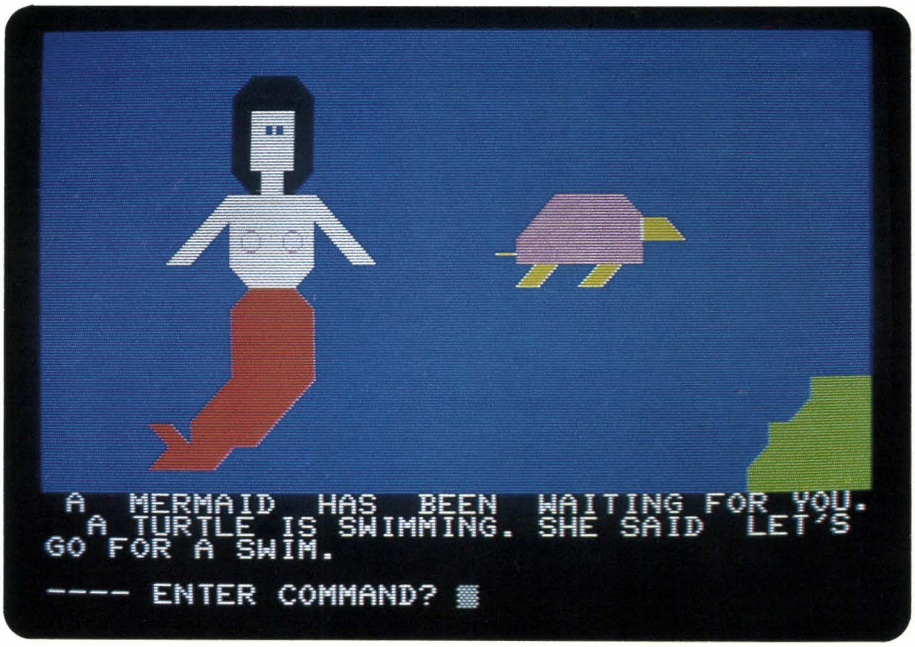
よーく考えて行動しないと

AUTOCRACYとは独裁政治の意味。と書くと，皆さんは，「さては，近未来SF？」かとも思うかもしれませんが，実はそんな中途半端なものではない。場所は南海（だと思う）の離島。そこには，イースター島のように，モアイ（石の巨像）が立っており，何やら神秘の島なのです…。

というわけで，BASIC HOUSEから発売中のMZ-700用のアドベンチャーゲームのテストランの報告です。ゲームはHu-BASICをロードしたあと，モニタからロードします。ゲーム中はすべて英語で，最初に出てくる作者名は，国籍不明。もっとも日本人じゃないことはたしかでしょうが（なんでも，MZ-700はヨーロッパにも輸出されているから，そこらへんかな？）。どうも最初から謎ばかりですね。中身は本場の英語らしく，おそらく高校級の英語力が必要と思われます。少なくとも，このゲーム中には，英語辞典は手離せないでしょう。

さて，このゲームの目的は，あなたの知恵と行動力で独裁政治を打破することです。といっても，チョビ髭をはやした，「いかにも！」のオジサンが出てくるわけではありません。実を言うと，このゲームには，人間はまったく出てこない！　いったいだれが独裁政治を行なっているのかさえもが謎となってきています。

さて，ゲームを開始すると始めのシーンは夜。まあ，夜ですから，寝ようということで，SLEEPと入力。あっという間に夜が明けます。そこからは島をさまようことになるのですが，これはほんのはじまりにすぎないのです。実は，ここだけの話ですが独裁政治は，地上で行なわれているわけではないのです。ですから早い話，地上で多少行動したのちには，別のところへ移動しなければなりません。はっきり言って海中なのです。さて，海中でいったい何が必要になるのかを，よーく考えて行動してくださいね。その後はもう海底をさまよいながら苦労するしかありません。もう少しだけヒントを出しておくと「見かけで善悪を判断してはいけない」ということと，「悪は滅びなければならない」ということです。

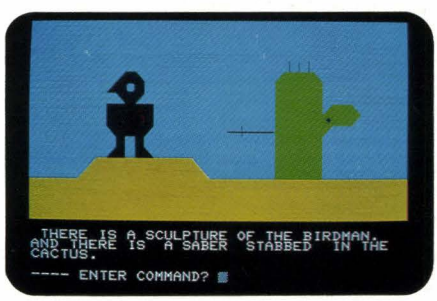


編集室が受けとったのは，サンプル版で，マニュアル等もまだ制作中とのことでした。最終的にどのようなものになるかは，まだ不明なのですが，このゲームに良いマニュアル（ヒントもある程度入れたほうが良い）が付けば，おそらくMZ-700用のアドベンチャーゲームとしては，かなりのものになるのではないでしょうか。画面もキャラクタ単位なのに，十分手がこんでいますし，描画速度も十分速いですね。アドベンチャーゲームでは，描画速度も大事なことです。イライラしないですみますから。

ただし，遊んでみて困ったのは類義語をあんまり受けつけないことです。たいていのアドベンチャーゲームでは，取るならGETかTAKEでよいのですが，このゲームの場合は，もうひとつ考えて単語を選ばなければなりません。

とにかくも，MZ-700用の本格的なアドベンチャーゲームとして，期待したいですね。

（高野庸一）



君はヒーローになれるか

「惑星メフィウス」のX1用が発売になったとき，思わず〝ヤッター!!〟と叫んでしまった。多くのX1ユーザーも同じであったに違いない。それほど〝メフィウス〟は高い評価を集めていたからだ。そして現在，すでにFM-7, PC-8801では第2部「暗黒星雲」が発売され，スターアーサー伝説の話題は最高潮に達しているのだ。その人気の秘密はいったいどこにあるのだろうか。

遂に姿を現わした■■■

スターアーサー伝説の第1部「惑星メフィウス」が最初に発表されたのは昨年の夏。FM-7用でカセットテープ3本組で4,800円パッケージの装丁からも当時としては，かつてない超大作といったイメージがあった。もちろんアドベンチャーゲームの性格上，発売と同時に大ヒットというわけにはいかなかったが，実際に挑戦したゲーム狂の友人たちの評価は絶大であった。そして11月，「暗黒星雲」の開発が進行しているという話題が流れ，〝スターアーサー伝説〟の構想が明らかになりつつある中，あせるX1ユーザーの前に，遂に「惑星メフィウス」はその姿を現わしたのであった。

物語はこうして■■■

〝スターアーサー伝説〟はアドベンチャーゲームとしては比類のないスケールを誇る壮大なスペースオペラで，「惑星メフィウス」はその序章にあたる。その物語は——

宇宙歴3826年，銀河連邦は外宇宙からの敵ジャミルと交戦中であった。凶悪なジャミルの宇宙艦隊によって壊滅状態となった惑星シークロンの英雄，スターアーサー・ミルバックはクランプトン2世号に乗って飛び立った。シークロンに伝わる伝説の剣〝レイソード〟を求めて，銀河辺境の惑星メフィウスへと向かったのである。

まずカセットの1本目をロードすると，いささか大時代的なタイトルにはじまり，つづいて今度はちょっとイキなスタッフ紹介がある。ナイス！　HIT ANY KEY！

税関でアンドロイド（女性カナ？）が目をピカピカとさせながらパスポートを見せるよう要求してくる。いきなり言われてもあるはずがないのだが，名前や目的などを答えさせられたあげく逮捕されてしまう。このとき，おもしろいことにスターアーサーが連行されるシーンが画面に描かれる。つまり，スターアーサーに扮する自分自身が捕まるシーンを見ることになるのだ。これはアドベンチャーのドラマ性やキャラクター設定の重要性を意識しているからだろう。

こうしていやおうなしに牢屋に入れられることになり，なんとか脱出しなければならないのだ。従来のアドベンチャーでは，○○ヲミル，○○ヲトルといったコマンドが基本であったが，このメフィウスには新しいアイデアが取り入れられている。たとえば，トルとかシラベルとかを入力すると画面にカーソルが現われ，目的のもののところへカーソルを動かして↵を押せばよいのだ。さらに，ゲームの進行上中心となるのは会話モードである。牢屋の中には2人の異星人がいて，この会話モードによっていろいろな情報を聞きだすことができる。そして，ここでの会話が脱出後の展開に大きな意味を持ってくるのだ。ただし，いいかげんなことを聞くと，「ちゃんとしゃべれよ」といってバカにされてしまう。

さて，脱出に成功すると1本目のテープは終了で，2本目に受け渡すゲームデータが出力され，これをブランクテープにセーブしておけば，以後は2本目からゲームを始めることができる（2本目終了時も同様である）。かなりめんどうではあるが，これだけの超大作をテープ版にする以上しかたがないことかもしれない。

牢屋のあとはスラム街で，多くの異星人たちが登場する。ここでも対話が中心で，できるだけ多くの情報を聞きださなければならないのだ。またスラム街から抜けてシティへ行くには，2通りの方法があることもわかるはずである。

シティでは気軽に買い物ができるが，なんといってもT&Eソフトなんていうのがあるのがいい。しかも，中に入ると横山さん自らも登場するのである。

そしていよいよ3本目，スターアーサーは広大なメフィウスの砂漠地帯をさまよい歩くことになるのだ。それはそれまでの道のりの何倍もつらくきびしい試練であり，私の友人も何週間と砂漠を歩きまわったそうで，かく言う私も何度も怪獣に食べられ骨になってしまったものである。砂漠はなんと22×22＝484のエリアに分かれていて，画面上に呼びだせるマップと説明書のマップをチェックしながらひたすらさまよい歩き，最終段階の宇宙ピラミッド内へと進まなければならないのだ。そして……

不屈の精神こそ■■■

T&E SOFTの作品にはデザイン的にセンスの良いものが多く，特に「メフィウス」のグラフィック画面には目を見張るものがあります。FM-7の移植版ですが，PC-8801とは異なり，X1用のグラフィックエディタで絵を最初から描き直したそうです。ただFM-7に比べペイントが大変遅くなっているのはちょっと残念です。本誌では現在，X1でもFM-7に負けない超高速ペイントルーチンが発表となり，次号ではさらに速い超光速(?)ペイントルーチンも発表の予定なので，X1だから遅いというのは少しガッカリするのです。もちろん，「メフィウス」のペイントとて水準以上のものであり，中間色の美しさも考慮すれば総合的に見て最高のグラフィック画面であることに変わりありません。

そして「惑星メフィウス」のもつ最大の魅力は，そのバラエティに富んだキャラクターにあるような気がします。STAR WARSをはじめ，ガンダム，マクロスといったアニメーションにおいても，それぞれのキャラクターのもつ魅力が作品の人気につながる場合が多く，パソコンのアドベンチャーもこのようなスペースオペラの形をとる以上，キャラクターの重要性は大きくなるでしょう。主人公のスターアーサー自身のキャラクターはまだまだ明らかにされてはいませんが，「暗黒星雲」ではスターアーサー自身にかかわる重要な要素が明らかにされるようです。しかし，あくまでもスターアーサーというキャラクターは，プレイヤーであるあなた自身が育てていかなければならないものでしょう。そのためには，あなたが「メフィウス」のなかでスターアーサーになりきることが大切なのであり，それが数々の難関をのりこえて伝説の剣〝レイソード〟を得るための最大のポイントになると言えるのです。

アドベンチャーゲームをいわゆる謎解きのゲームと考える限り「メフィウス」は解けないでしょう。いまだに牢屋から脱出できないあなた！　どこかにドアのカギが隠されているなんてさがしてもムダですヨ。古い牢屋に実際に閉じ込められたときにどうするかを考えれば，不屈の精神を持つスターアーサーにとって脱出は決して難しくはないはずです。

「惑星メフィウス」は，ドラマのスケール，難易度，人気，デザインともに一級の出来ばえといえます。そしてこの一大アドベンチャーを解いたとき，ようやく〝スターアーサー伝説〟は佳境へと入っていくのです。第2部「暗黒星雲」のX1用が1日も早く発売されるようT&E SOFTさんにお願いしたいと思います。

スターアーサーよ永遠に。　(斎藤　晋)

スターライト　アドベンチャー
STARLIGHT ADVENTURE PART I & II

X1/C/D, MZ-2200/2000

ヒーローを殺しつづけて

以前から一度は紹介したいと思っていた「スターライトアドベンチャー」。X1, MZ-2200/2000用ともに, PartIIまで発売され, 今月号はアドベンチャー特集ということもあって, ついに発表する機会を得られたのです。思えば私が初めてアドベンチャーに触れ, 最初に解けたのが「スターライトアドベンチャーPART I」でありました。というのも手伝って, とても感慨深いのでありますよ。まずはとくとお聞きあれ。

南に進むと■■■

私はMM。相棒の宇宙船Z号は, つねに優秀であり私の良きパートナーである。どんな辺境域でも彼女といっしょなら行っちまう。いわば, ハンソロとファルコン号の関係であろうか。もっとも彼らは映画出演の声がかかるほど有名ではあるが…。

今回私は, 8つの星を持つ太陽系にやってきた。さあ何が起こるか, 楽しみだぜ!

まず第8惑星アイリスに行くと, まだ早いと追い出されてしまった。どうもここは最後に来るところらしい。それでは, と第1惑星ウォーツから行ってみるか。

ここはウォーツ。良き友, Z号のコンピュータによると, 北と南に道があるという。さて, 北に行ってみると, 洞穴があった。中に入ってみたが真っ映で何も見えやしない。明かりがいるかといわれても, ライトは持っていないでしょ, Z君。とりあえずライトがいりますね。しかたがないから, 南に進んで行くと, 困り果てたウォーツ人に会った。よっしゃと相談を受け, 次の惑星オーソニアへと私の旅は始まるのであった。

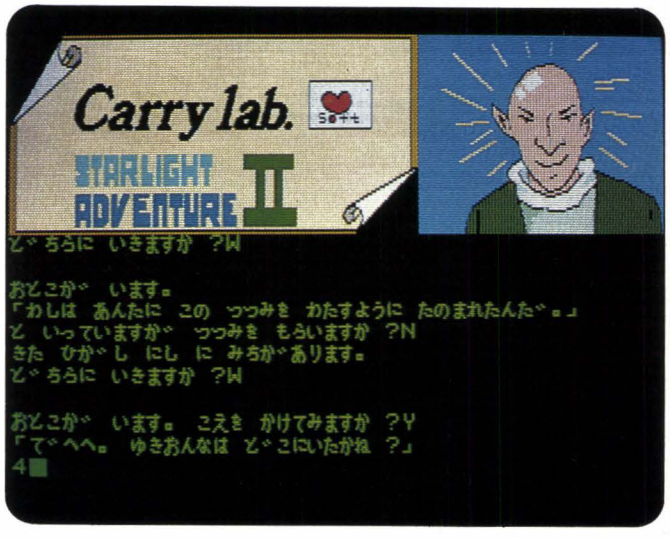
行き詰まるところは■■■

テキスト, グラフィックを問わず, アドベンチャーゲームの多くは, 単語を入力することによって進行していきます。つまり, 名詞(～を)＋動詞(～する)の形で, 日本語または英語を入力していくのが一般的ですね。ところがこれには限度があり, 同義語が使えない場合がままあるのです。物を取りたいとき, GETでだめで, TAKEでうけつけられたり, また, カギのかけてあるドアを開けるときにも, 〝カギをつかう〟のではなく, 〝カギをあける〟であったりするのです。矛盾を感じますね。これには, メモリの容量, または作者のvocabularyの貧困さなど, 種々の事情があいまって, しかたのないことかもしれません。

結局, アドベンチャーの極意というものは, 自らの推理により純粋に未知なる謎を解いていくところにあるのだからして, 同じ行動に違った言葉を選ばなければいけないというのは何たることか。と, つい怒りの声をあげてしまいましたが。かの歴代の名探偵諸氏が, 繊細かつ大胆な行動のうちに, ここはカギを「つかう」べきか「あける」べきかなどと迷ったでありましょうか。否であります。ポワロ然り, ホームズ然りであり, フィリップ・マーロウに至っては, まあ形容詞ぐらいには気を使ったかもしれないが…。

そういう意味も含めまして,「スターライトアドベンチャー」はすべてYes, No方式を採用し, 行動する主人公に指示をあたえていく形で謎解きのみに専念できるのです。生きるの死ぬも私次第。何度, ヒーローを殺してしまったことか。行き詰まるところは理解の足りない自分であるから, かわいさあまっても, 憎みきれなかったりして。

PART Iは,内容を考えても,小手調べとして初心者の方にはうってつけでしょう。



謎が解きかけたときの,「あっなるほど,そうであったか」と感じるあの興奮と快感を味わっていただきたい。そうしてより難解なアドベンチャーへとのめり込んでいくのです。

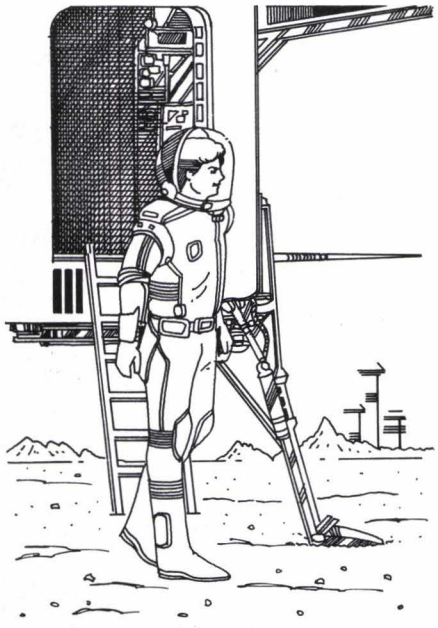
新たなる冒険に■■■

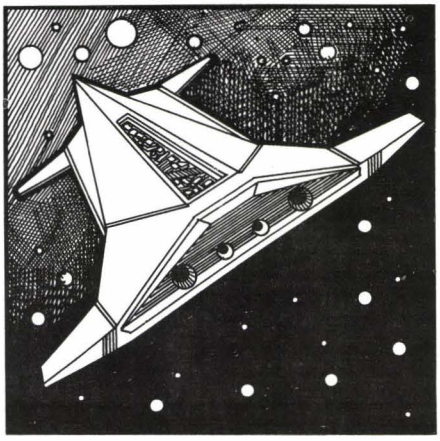
私MMは,紆余曲折,右往左往ののち,マスターにやっと会えた。が,これが次の冒険へとなるきっかけになるとは,思いもよらなかったのである。

アイリスに着いたとたん,マスターが撃たれ,シャンが連れ去られてしまった。とにかく探し歩くとするか。しばらくすると包みを持った男に会い,包みをくれた。これを開けると…。しまった!

乱数とマッピング■■■

謎を解いていく上で必要なことに,マッピングがあります。 これは,どこに何が





あり,誰がいて何をするかをすべて書き入れていき,どこで死が待っているかなど間違いを繰り返さないためにも,ぜひ作っておきたいものです。私が独自に作っているのは,紙面に東西南北を決め,出発点を中心に動きまわるごとに自分のブロックを描き,出来事をブロック内に書き込むということをしていますが,こればかりは,個人のやりやすいようにマッピングするほうが,おもしろいですね。

ところが,時には役に立たないこともあるのです。それは,プログラム内で乱数が使われている場合で,毎回プレイするごとに出てくるもの,道などが違ってくるからです。これは,ゲームを難しくするために使われる手法のひとつなのですが,使い方によっては目もあてられないものになる恐れもあります。以前,あるアドベンチャーで,やはり乱数を使っており,怪物の出てくる場所や,ラックに入っている道具などがそのつど違って出てくるので,やたらと死にまくるというのがありましたが,これは仲間うちでもひんしゅくを買っておりました。

ところで,PART IIでは一部分だけ乱数を使っており,これなどは成功しているほうではないでしょうか。考えてみれば,いつも同じ場所に同人物がいるわけはなく,ちょっと軽薄な男がうろつきまわっていても何ら不思議はないのです。まあ,ゲームの性質上,すべてがそうではこまりますが…。

彼の出方によって,ゲームの進め方を臨機応変に講ずるのも楽しいものです。だからというわけでもありませんが,PART Iに比べPART IIはやや難度が増しているようです。同じく入力は,Yes, No, 方式を取っており,人物の出方,消え方などはよく考えられ,続けてアドベンチャーを解いていくには,良いものでしょう。

PART IIからでもゲームはできますが,やはりPARTIを解いてから挑戦したほうが,より楽しめると思います。PART I, IIともにX1用, MZ-2200/2000用が発売されてま

して,グラフィックの点では,X1とMZ-2200はさほど違いはありません。が,おもしろいのはMZ-2200で,PSGは,1音しか出ないはずが,メロディーが和音に聞こえてくるのはすごい。一聴の価値はありますよ。

まだ根はあげられぬ■■■

なんと包みは爆弾であったか。同じしくじりをしないように歩きまわったあげく,出発点にもどってみると,ここは宇宙空港。今まさに,謎の第9惑星RANDONに飛びたつところであった。まだまだ根はあげられないぜ。 (三宅流衣子)

野趣あふれるグラフィック

「黄金の墓」は，ストラットフォードC.C.C.が主催した第1回アドベンチャーゲーム・コンテストにおける最優秀賞受賞作品である。制作者は，井上潤さんと大浦由貴さんで，オリジナルはPC-6001。本誌では昨年10月号において，同コンテスト優秀賞受賞作品となった「ピラミッドの謎」をカラー画面紹介し，好評を得ているが，このときの最優秀賞である「黄金の墓」がX1にも移植され，アドベンチャーゲームファンにはいっそうの興奮が期待できるはずだ。

砂漠をさまよい■■■

砂漠の謎は，真夜中に一枚の手紙が舞いこんだことから始まる。

「わたしは　クフおうを　しっている」。手紙にはこう書かれていたのだ。

「じつは　クフおうは　いきていたのだ。
そしてわたしに　タカラの　ありかを
おしえてくれた。しかし……」。

そして私は，焼けつくようなエジプトの砂漠に立っていたのだ。

さてゲームを進めるには，〝～を見る〟〝～をとる〟というように動詞と名詞を組みあわせて，ひらがなで入力すればよく，スタート時には，150のパワーと150の水，それに100のお金を持っている。パワーと水は移動するごとに減っていき，20以下になると力つきて死んでしまうので，あまりうろうろとしているわけにはいかない。とくに水は砂漠においては大切で，ときにはお金の代わりに水を要求されることもある。なんとかして補給しておくことが肝心だ。クフ王の宝を見つけるまでには長い道のりがあるのだ。

砂漠をさまよっていると，謎めいた人物が次々に登場する。すべてを見通したように語りかけてくる占師，正体不明の男ロイド。いずれも謎を解くカギとなる重要なキャラクターである。なかでもオアシスに住む美しい女性ネイトは，唯一の女性キャラで，疲れきった冒険者にとって悩ましくも魅力的である。ただし女性に対してはあくまでも優しくしないと[1]……。

何度も何度も殺されながら，各地点をくまなく調べていくうちに，全体のつながりがわかってくるはずであるが，それでもまだまだ納得のいかない不可解なことは多い。アドベンチャーでは地図を描いて進むのが定石であるが，ここではその地図までが，描き上げるにつれミステリアスなものになっていく。歩き過ぎて方向感覚まで狂わされてしまうという設定なのだろうか[2]。

こうして，とにもかくにも私の前に現われたのは，エジプトものとは思えない得体の知れない建造物であったのだ[3]。

ミステリアスな要素が■■■

「黄金の墓」はピラミッドやスフィンクスといった冒険心をかきたてるエジプトを舞台に展開し，ミステリアスな要素がいっぱいだ。特筆に値するのは，アップル風とも言える独特のグラフィック画面である。けっしてPC-6001からの移植だからあらいのではない。いわゆるアニメのセル画のような絵とは根本的に異なり，異国情緒を感じさせる見事な出来ばえだ。

問題点はむしろストーリーにあり，不合理に思われることや，脈絡のない展開も平気で起こってしまう。ただしメッセージはいいかげんなものも楽しいが。（斎藤　晋）

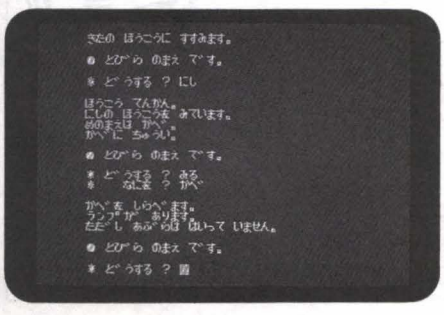
注）

1）やはり女性をくどくのにあせってはいけない。私の場合，なぜか何度も浮気をして彼女に殺されてしまった。
2）善意に解釈しているのだが…。
3）クフ王の宝の謎をムリやり解いてしまうカギがここにあるのだ。

魔王との闘いに備えて

聖なる剣は魔王にさらわれたお姫様を救出するために，魔王の洞窟へ乗り込んでいくひとりの剣士の物語です。ゲームの開始時点では剣士(あなた)は武器を何も持たず魔王の洞窟の前に立っています。やがて来るべき魔王との対決に備えて，あなたは何をしなければならないのでしょうか。なお，この聖なる剣は絵の無いテキスト形式のアドベンチャーゲームですから，自分の周囲には何があるか常に注意しておく必要があります。

まず最初に行きつくのがドラゴンのいる広間です。このとき，早くも最初の難関に突き当たってしまいます。このドラゴンをどうにかしないことには，広間の他の場所へ行くことすらできません。鎖につながれているからといって移動しようとして，ドラゴンのツメにかかってあっという間に死んでしまった人も多いのではないかと思います。ドラゴンを何とかした後は，武器（特に剣）を探さねばなりません。このためには，かつて魔王に倒された戦士のメモが重要なヒントになります。このメモを読まなければ，それがあまりに突飛なことなので，剣を手に入れることはまず不可能でしょう。このメモがどこにあるかは，ちょっと教えられませんが，次のようなことは言えます。棚や箱の中からある物を取ったとしても，後には何も無いなどとは思わないこと。



ドラゴンのいる広間からいろいろな部屋に行くことができますが，多くの小道具を手に入れることができる割には，誰とも出会うことができません。これでは何も解決しないではないかと思っていると，意外や意外，地下室に通じる秘密の通路を見つけることになるのです。そして，このゲームの本当の面白さは，この地下室へ降りてきてからにあると言うことができます。ところで，地下室を移動するには，ランプが必要です。この時点までに，ランプ，油，火打ち石は見つけておきましょう（私は油を探すのに苦労してしまった)。

地下室ではいろいろな怪物との出会いが待っています。このとき，それまでに集めた小道具をうまく使って怪物から逃げ出さなければなりません。しかも，５回の命令以内に，という条件が付いています。これは結構きつい制限で，もたもたして殺されてしまうということも多々あります。こういう状況は，血に飢えたドラゴン，食肉植物，殺人バチ，ミノタウロス，吸血鬼，毒グモ，ヘビ女と，ひと部屋進むごとにやってきます。これは，もうスリルの連続で，手に汗握る展開です。それらの難関を切り抜け，やっと，目指す魔王の部屋へと出ることができます。そして，最後の難関，魔王直属の部下，黒騎士との対決が待っているのです。それに勝利すると，あなたはお姫様に会うことができます。しかし，お姫様は，魔王の魔法でビンの中に閉じ込められていますから，どこかに隠れている魔王を探しだして，それを打ち破らねばなりません。魔王との一騎打ちに際してどうしても必要なのが，このゲームのタイトルである聖なる剣です。したがって，それまでに聖なる剣を手に入れていなければなりませんが，これはかなりの難問でしょう。とにかく，魔王との対決で聖なる剣を使うまではいいのですが，その先がちょっとわかりません。ここでも５回の命令のうちに魔王を倒さなければならないのですが，かなり苦労することになります。しかし，ゲーム開始時のメッセージをよく思い出すことで，見事に魔王を打ち倒し，ハッピーエンドを迎えることができます。

（清水和人）

宇宙船は故障していた

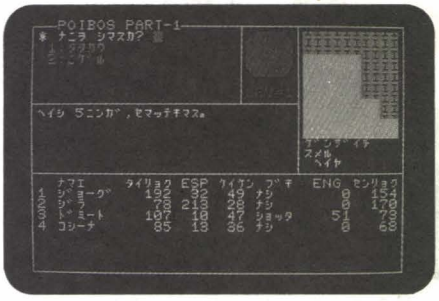
俺の名はジョーグ。この宇宙のどこかにいるポイボス七聖人のひとりだ。今は，宇宙船ゼピュロス号の機関士をしながら，遠ざかっていくクラーネ星を眺めている。俺達の母なる星ポイボスはクラーネ星の手によって宇宙から消滅してしまった。失なわれた故郷を再建するための旅，第二のポイボスとなるべき惑星を探す旅が今始まったのだ。

しかし，思えば大きな代償だった。俺達のクラーネ星脱出のために多くの仲間の命を戦渦の中に散らせてしまった。ヒャア・エフ，テツノスケ，グーリン，ヨカナーン，コシーナ，君達の尊い犠牲は決っして無駄にしない。

リュキア，ミネル，君達もどうやらポイボス七聖人のひとりらしい。お互いに力を合わせて残る四人を探し出し，一日も早く第二のポイボスを見つけよう。

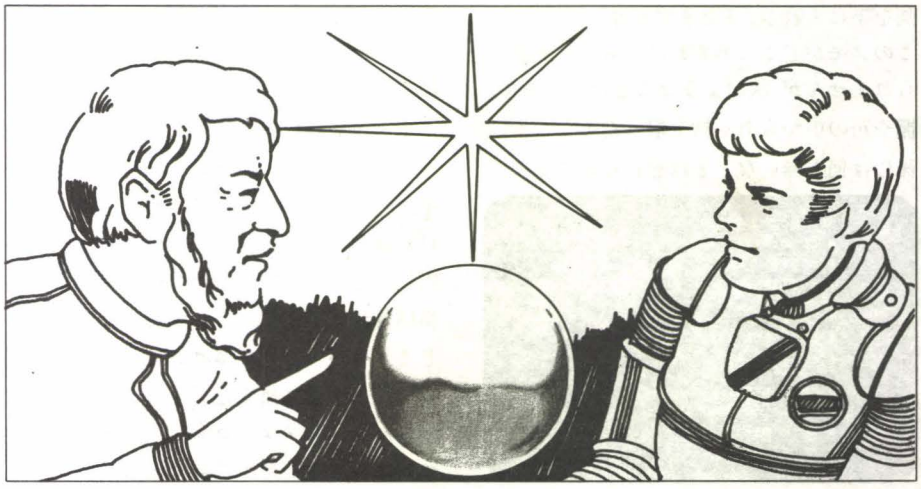
ゲームが終わってから，思わずこうつぶやいてしまいました。それほどまでに主人公に対する感情移入をさせてくれるのが，このロールプレイングというものなのでしょうか。とにかく，ゲームを開始したばかりのころは，戦闘の仕方など要領がなかなかつかめず，なんでこんなものが面白いのだろう，という感じもしましたが，数時間も遊びつづけると，もう夢中になって途中で止めることができなくなるゲームが，このポイボスPart Ⅰです。

物語の発端は，消滅したポイボス星からかろうじて脱出した人々が捕えられているクラーネ星の収容所のひとつ，ジルダ＝3です。主人公ジョーグはこの場所から，今はまだどこにいるかわからない，ポイボス七聖人やクラーネ星脱出を手助けしてくれるはずの仲間を求めて，クラーネ星を彷徨することになるのです。ジョーグの行く手には様々な敵が待ち構えています。民間人，監察官，ロボット，兵士，機甲兵士などの攻撃を回避しながら，あるいは彼らと戦闘しながら，仲間を増やしていかねばならないのです。



このゲームの特徴として，登場人物の多彩さがあげられます。なにしろ10種類の異なる星の人々が登場するというのですから，ある人物に出会うごとにマニュアルの説明を読んで，この人種はこういう性格を持っているのか，などと考えながらゲームを進めていけば面白さが倍増するに違いありません。私の場合は，ゲームを終了するまでに23人の人物と出会うことができました。実際にはいったい何人に会うことが可能なのでしょうか。人に出会うと，その人を仲間にするかどうかが問題となります。敵と戦うには仲間が多いほうがよいのですが，やみくもに会う人すべてを仲間にしていくと，手ひどいしっぺ返しを受けることもあります。たとえばある人物を仲間にすると，全員の武器を持って逃げられてしまいます。また，結局はこちらを裏切ってしまうのですが，強力な武器を持って来てくれる人もいます。こういう人に対しては，仲間に入れた後，すぐさま武器配分を行なって，その武器を自分のものとしておけばよいでしょう。

ところで，ゲームのマニュアルに付いているクラーネ星帝都市マップはかなりいいかげんです。私は，アドベンチャーゲームの癖で正確な地図を作ってみましたが，こういう作業もこのゲームの楽しみのひとつでしょう。もっとも，舞台のすみずみまで隈無く行こうと思うと，それまでに何度か死ななければなりません。たとえば，エネルギープラントの辺縁地へ行くと熱射病で死んでしまいますし，廃液湖の側では廃液ガス中毒にかかったりもします。



しかし，このゲームの最大の面白さは敵との戦闘にあると言えます。敵が迫って来た場合，戦うべきか逃げるべきか，というのが最初の問題です。機甲兵士やオーガノイドを相手にする場合，逃げ出す時期を誤ると，どこまでも追いかけてこられて，最後には全滅ということにもなりかねません。敵と戦うときは，味方と敵の戦力の分析が重要になります。この戦力分析というのは，1から4のレベルがあります。このレベルは，今までにどれだけ戦闘をしてきたかという経験によって決定されるものです。レベル1やレベル2では正確な分析はできませんが，レベル3，レベル4になると敵や味方の真の戦力がわかりますから，確実な攻撃をすることができます。この点で面白いのは，出会ったときの敵の戦力が，味方の戦力を上まわっていても，戦闘中に逆転することがあるということです。ポイントは，敵のリーダーを1回目の攻撃でやっつけるということでしょう。いくら敵の力が強大といっても，こちらの体力が十分残っていれば，1回ぐらい攻撃されても死ぬものではありません。かなりの強敵と言われているクラーネ軍の兵士の6人部隊などをうまく撃破したときは快感そのものでした。

このゲームにおいて，大きな戦闘は，ナネー海軍基地での戦闘とクラノレ主都圏での戦闘でしょう。このふたつの地区にはとても重要な人物がいますから，ゲームを終わらせるためには避けては通れない道になっています。当然，そこでの戦闘に勝利しないことには目的の人物に出会うことはできません。しかし，そこの警護の厳しいことといったら，他の地域の比ではありません。なにしろ，オーガノイド，小隊，兵士，機甲兵士といった連中がウジャウジャいるのですから。このため，それまでに強力な武器と体力を十分蓄えておいて決戦に臨まなければならないのです。それも，絶対にかなわない相手なら逃げなくてはなりません。無駄な戦闘をして武器のエネルギーを減少させることはできないからです。しかし，何とか勝てそうだという敵に対しては果敢に攻撃を仕掛けましょう。戦闘中に仲間がひとり，またひとりと死んでいくこともありますが，ポイボス人が殺されない限りは戦い続けることができるでしょう。ポイボス復興のために，多少の犠牲はやむを得ないのです。

クライマックスの戦闘が終わると，脱出用宇宙船の隠されている場所がわかるようになっています。ここでゲームは一直線に終結へ…と思ったのですが，そう簡単には終わりません。宇宙船は故障しているのです。そこで，敵の攻撃をかわしながら，宇宙船を修理することになります。せっかく直りかけた宇宙船が，敵の攻撃で被弾し，修理にかかる日数が増えてしまうこともありますから注意しましょう。

そして，めでたく修理が終わると，パイロットと機関士を決めて宇宙船の発進です。このとき，長かった脱出への物語は終わりPart IIへ向けての新たな旅立ちが始まるのです。Part IIではどのような試練が待ちうけているのでしょうか……。（清水和人）

調書の作成がカギです

この「ソフトハウス殺人事件」は「コスモクロス」「聖なる剣」でお馴じみのクリスタルソフトより4月21日発売予定という，できたてのアドベンチャーです。

LATXソフトというソフトハウスで起きた殺人事件を，あなたが探偵クマダになって解明していくという設定です。

やはりこういう事件の捜査は聞き込みが基本。ソフトハウスを中心として，現場付近を歩き回り，登場する6人の人物たちに話を聞いて回らなければなりません。とりあえずはLATXソフトに行って現場を見せてもらおう，と思って入りましたが，最初の登場人物である社長，勝手に動こうとすると怒り，なぜか現場に通してくれません。う〜む，話のもっていき方が悪かったのかな，というわけで，もう1度社長とじっくり話してみることにします。

という具合いの，聞き込み中心のアドベンチャーゲームですが，これもアドベンチャーゲームの鉄則にもれず，取るものはすべて取る，聞けることはすべて聞く方式でいって間違いなさそうです。ただし，やっていてわかってきたのですが，最初聞いたときには「知らないよ」などと言っていても，他の人への聞き込みによる捜査の進展によっては，次に同じ質問をしたときの答えが違ってくることがあります。よ〜く推理しながら，再度質問を繰り返すのがこのゲームのコツといえそうです。しかし，都合の悪いことを聞かれると雲がくれしてしまう人もいますから，聞き方には十分注意しましょうね。

さて，入力方法はカナ入力で，慣れないうちはちょっと戸惑いました。まあ，制限時間に近づくころにはかなり慣れるものです。また，入力できるコマンドはすべて公開されており，隠しコマンドは一切ありません。

画面処理のスピードも結構速く，面数こそ少ないものの，それなりに工夫されているようです。

よくあるアドベンチャーゲームのように，何か行動すると突然殺されてしまったり，死んだりというようなことはありません。が，時間制限があります。朝の10時に捜査を始めて夕方の6時ぐらいまでがリミットの時間で，行動をひとつ起こす，または聞くごとに2分が経過します。「なんだ240回も入力できるじゃないか」などと甘くみてはいけませんよ。私もこの事件の犯人を見つけ出すまでに，3回も時間オーバーしてしまい，犯人に逃げられてしまったのですから。とにかく，あまり無駄な行動，質問を繰り返していると，あっという間に6時をまわってしまいます。

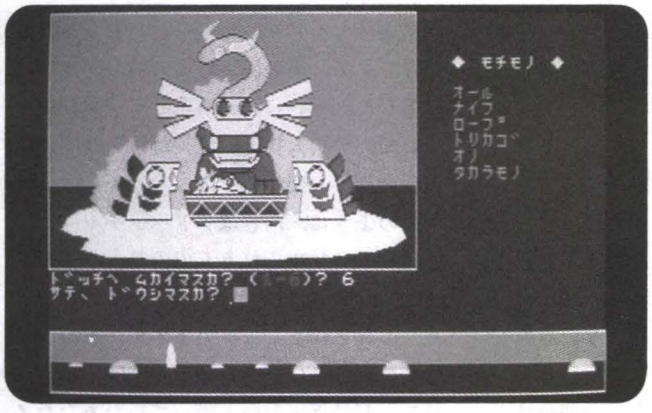
このゲーム，殺人事件という名に恥じず，推理を十分働かせられるよう工夫されている点は見事です。あるものを手に入れるために満たしていなければならない条件，というアドベンチャーゲームの定石をうまく利用してストーリーを組み立てているところも，推理ゲームとして十分な手応えがあるといってよいでしょう。

最後に，移動型アドベンチャーで，よく地図を描いて自分の位置を明確にするように，このゲームではそれぞれの登場人物ごと，あるいは時間ごとに，自分なりの調書，メモを作っていくと，のちのちの推理の参考になるでしょう。画面上だけでなく，紙の上で考えることでよい結果が得られそうです。

このタイプのゲーム，1回解いてしまうと犯人がわかってしまっておもしろくなくなるものですが，2回め以降はいかに早い時間に解決できるかを試してみるというのも楽しいのではないかな？　（谷川展子）

持ち物に制約はありません

グラフィックがさらに,細かに,そしてカラーが常識となった今，パソコンはワンダーランドとなった。いくら計算機とはいえゲームのできないマシンなんてつまらないもんね。大型計算機の上でさえ，端末を操作してスタートレックなどキャラクタゲームで遊ぶこともある。グラフィックもできないし,TSS処理だから反射神経型はまず無理。つまり思考型のものしかできない。大型計算機は処理が早いっていうけど小回りがきかないからね。



その点パソコンはすごい。すきな時に好きなものを遊ぶことができる。しかもランニングコストはタダ。今ブームのアドベンチャーゲーム，一昨年のMYSTERY HOUSEに始まって，T&Eのメフィウス,で現在の戦国時代となった。毎月発売される雑誌に数多くの広告が掲載されているが，画面の美しさは新しく発表されるほど良い。専門のデザイナーがデザインするものもあるというから驚きだ。

私はミステリーハウス（マイクロキャビン）のころからアドベンチャーゲームに熱中した。連日数時間ずつナゾ解きを楽しんだ。その後パート２に裏切られたために(あんなのゲームとは言わん）それ以来遠ざかっていたのだが，今度久々にアドベンチャーゲームに挑戦することになったのである。

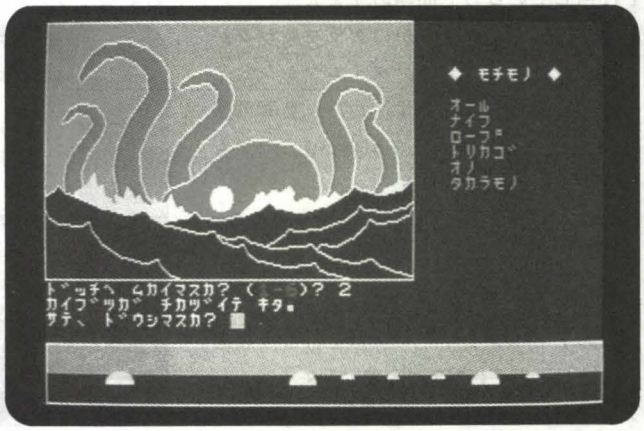
ヤッパ，わくわくするね，パッケージを開く瞬間っていうのは。開けてびっくり，dB-BASICで走るとある。バージョンの指定は特にないみたい。そこで古いものよりは，ということで最新の1.2を起動する（テープは遅いなあ！)。続いてゲームプログラムをロード。ナニ？　コンパイラしても動かない？　理由は簡単。プログラムを分割ロードし，一部（多分データだと思う）がマシン語のため。BASICのリストも何ら手を加えられていないので，そのまま見れた(でも,先を見てしまうと答えがわかっちゃうからグッとこらえてネ)。

会話はカナ入力。ちとやさしいように思えましょうが，そんな世の中甘くない。マニュアルを見ても何ひとつキーワードがないからね。唯一のたよりはファンクションキーのみ。メチャクチャに押して見れば何とかなるもの。あっそうそう，ストーリーを書くのを忘れてた。ゲーム自体の物語を全編と考えた場合，序章っていうところかな。つまり，UCLAの学生がミクロネシアへ隠し財産を捜しに行こうと，おばの遺産をすべて処分し，飛行艇をチャーター，パイロットと共に出発するが，嵐におそわれ不時着，遠のいていく意識の中で飛行艇の爆発音を聞き，パイロットのいないことに気づかないまま気を失った——。

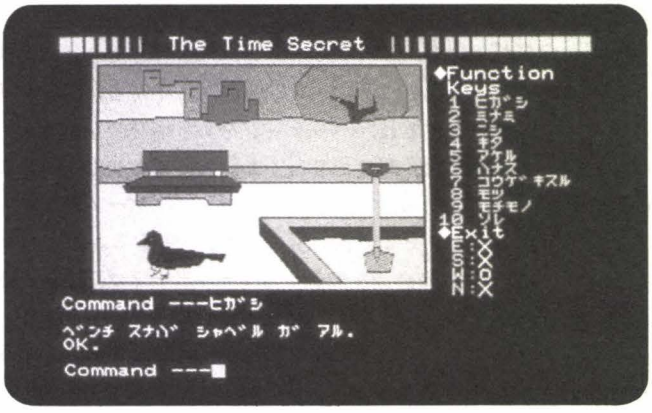
ゲーム開始時は彼が気がついたときからになる。目の前には島があり,何だか鳥かご，ナイフ，ロープがある。何か役に立つだろうと思いすべてひろって次の島へ。方向別に６つの中から選択できるけど最初は１だけみたい。後で何か支障が生じるだろうけど，数個という持ち物の制約がないのは不思議。こんなに持ってよく歩行できるなあ。

このゲームをする上でもっとも重要なことは出てくるキャラクタに対しむずかしく考えないこと。神前のおそなえなんかに手を触れようものなら即ゲームオーバー。ゲームの難易度は初～中級。気軽に楽しめるもので，１時間もあればクリアが可能。キャラクタはあまり美しいとはいえない。穴に落ちたとき底にある物が何だか理解できなかった。メッセージで表示するなりしないと致命的。１コマンドごと１面描きなおすのはイライラする。これは部分で処理しなかったのだろうか。

(松崎　洋)



「清水和人」氏の助けを借りて



MZ-700用としてすでに有名なアドベンチャーゲーム「タイムシークレット」がMZ-80K/C/1200, X1用に移植され販売されることとなりました。

最近はソフトの出がめっぽう少なくなったMZ-80K/C/1200ユーザーにとって，涙が出るほどうれしいことではないですか（なにをかくそうこの私もMZ-80Kのユーザーなのです）。

「タイムシークレット」といえば，今年の3月号THE SOFTOUCHで紹介されていますが，その素晴らしさは確かに書かれているとおりです。

MZ-700用とMZ-80K/C/1200用とはどこが異なるか？　というとハード的にはまずカラーが出ないこと。エレベータで下に降りるときのライトの点滅とか，エメラルドなどの表現がいまひとつパッとしないのが残念です。それからファンクションキーがありませんが，右側のテンキー(?)の部分で代用しているので問題ありません。

また，X1用ではさすがにグラフィックがきれいに使われており，タイムマシンなど各所で見ごたえ十分です。ただ，音響効果は，なぜかMZ-700のほうが良いような気もしますが(写真はX1用)。

ゲームの内容はほぼMZ-700用と同じですが，ごく一部（たとえばロックナンバー）が違っているようです。

このアドベンチャーゲームを行なうにあたっては，すでにMZ-700用の「タイムシークレット」を解いたという超人的な人物，清水和人氏の助けを借りました。というより，私は彼の言うとおりにキーを押し，その手順を紙に書き留めていっただけなのです。

それでも，1時間以上はかかったでしょうか。初めて行なう人でも納得できる程度の最短コースを通ったわけですが，なんと手順を書き留めるのに使った紙はA4版が7枚。どんなに少なく見積もっても300以上の命令の文字を入れなければならないわけです。

清水氏いわく，「これでも会話の部分は省略したんだよ」

こんなすごいのを彼はどうやって解いたのでしょうかネ？　友人とで2カ月ほどかかったということですが，とんでもない場面で思いもよらないことを行なわないと，決して解けるものではないのです。MZ-700用をやったことのある人ならわかるでしょう。あの十字架の前で×××するとか，それを××××にわたして通る，なんてことはなかなか思いつくものではありません。

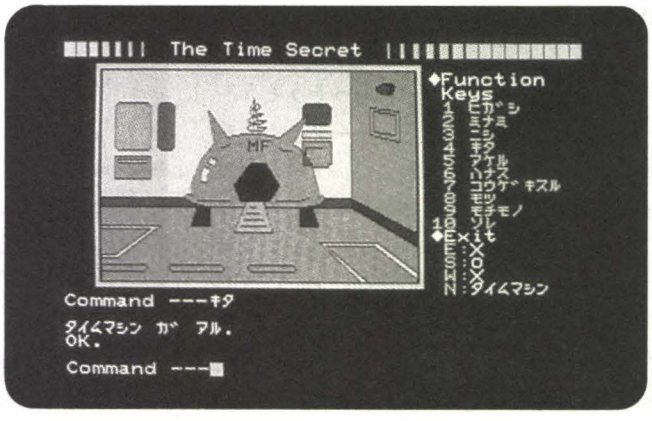
彼に言わせると，答えを連想させるヒントがどこかしらに出てくるのだそうで，あとは努力と根性，それと想像力をはたらかせてあらゆる可能性を確かめていけばよいわけです。

私はほとんど頭脳労働を行なわなかったわけですが，奥の深さに血の気のうせる思いでした。それに，結構余計な枝葉が付いていて，登場人物との直接解法とは関係のない会話でも楽しめます。

このように，内容はアドベンチャーゲーム狂に十分勧められるもののひとつです。

MZ-80K/C/1200を持っている人はひっそりと身をかくしてはいますが，まだまだ相当数いるわけです。K/C/1200用の新作ソフトが最近出ないと嘆いていたあなた！　この機会にもう一度MZをよみがえらせてみませんか？　また，X1ユーザーのみなさんも。そして，「タイムシークレット」の世界に足を踏み入れてください。ただし，その世界から出られなくなっても，当方はいっさいヒントをお教えしません。なお，この記事は自動的に消滅……，はしませんネ。

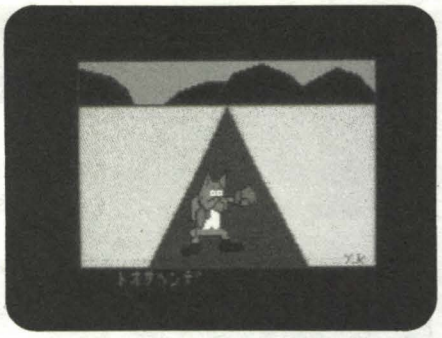
（大西　太）



ペンギンビレッジ

XI/C, MZ-2200/2000

んちゃ！　悩んでしまうのだ



アドベンチャー型といっても，これはいわば番外編。難しく考える必要はない。

もっとも，どうせペンギン村に常識などないので，まともにやるとバカを見る。なにしろ……ほよよ。

パッケージはもちろん，フタをあけるとアラレちゃんがオハコンバンチハ（切りとってパスケースに入れよっと）。テープをロードさせると，うほほ～い！　タイトル画面にも鮮やかなグラフィックでアラレちゃんが登場。ワクワク，ワクワク！　というわけでGAME START！

物語（というほどでもないが）は，トリ山さんの誕生パーティーに招待されたのだが，トリ山さんちはとっても田舎で，途中には分かれ道もたくさんある。そこで私はしっかりと道に迷ってしまうという筋書きなのである。ゲームの進行は非常に簡単で[A]～[L]と[5]，[7]，[8]，[9]キーに必要なセリフとコマンドが用意されている。ただし，道順には乱数が使用されているようで，道を覚えてもムダ。そして登場する動物たちがまたいんけんだったりして……。

さて非常に残念なことに，ゲーム中にはアラレちゃんも，ガッチャンも，せんべいさんも出てこない。もし私がアラレちゃんだとしたら，どうしてイヌとボクシングをして負けてしまうのか？　悩んでしまうのだ。

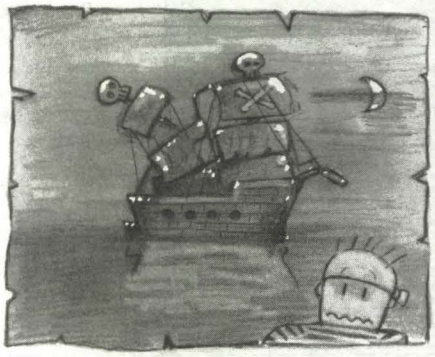
それにしても，今月は難解なアドベンチャーがひしめく中，気軽に楽しめる唯一のゲームではないだろうか。友だちと交互にためすのも楽しいだろう。

ところで，トリ山さんちは，なんであんなにりっぱな屋敷なのだろうか。またまた悩んでしまうのだ。　（斎藤　晋）

幽霊船

XI/C/D, MZ-2200/2000/80B, MZ-700

プログラム設計にただ脱帽



難しい。同じアドベンチャーゲームといってもこれはテキストアドベンチャー。つまり，絵が出ません。コンピュータが表示するメッセージに対しての応答で進行するゲームなのです。同種のものとして米国などコンピュータ先進国では昔から愛されているゲームの一種なんです。今でもZORK Ⅰ～Ⅲ（Infocom）がTOP10に顔を出すように親しまれています。しかし，難しい。先に進めないんです。甲板から通路，通路から部屋を歩き回り，駆け回って，タルやチョゾウコ，タイホウやガイコツに出会うのですが，あまり走りまわるとドアにぶつかって「イタタ！」ということになります。

そうです。事故にあって太平洋を漂流していたわたしは，目の前に現われた一隻の帆船によじのぼり，この帆船を探検するのです。この帆船が幽霊船であるとも知らずに……。

帆船の構造についてはきわめて詳しく，平面，立体の多方にわたって図解しておかないと迷路ゲームになってしまいます。どのアドベンチャーゲームもそうですが，資料片手でメモを取りながらゲームしないと頭の中でゴチャゴチャ。そのうち甲板から落っこちたり，足をすべらせて死ぬ，すなわちゲームオーバー。しっかり腰をすえてひとつひとつナゾを解く，これがアドベンチャーゲームの基本です。面倒だと思っているとこの世界には入れませんよ。

この幽霊船，状況設定といいアルゴリズム等抜群の設計です。プログラムはS-BASICで作られており，この種のゲームを作成したい向きには〝必修〟のソフトであるといえるでしょう。リストを見ても何ら解答につながらないデータ形式で収め，ルーチンそのものも簡素にまとめられています。解析に時間を要す練られた設計は，ただ脱帽のひとことに尽きます。

はじめにもいいましたが，初心者にはちょっと難しいかもしれませんが，試行錯誤してプレイを繰り返すうち必ず解けます。私はもう十数回「イタタ！」。メモを取りながらでもこれです。ヒント集が別途用意されているようなのでこれを手に入れてでも何とかクリアしたい。何となく先は見えてきたような気がします。　（松崎　洋）

ADVENTURE，アドベンチャー28選

ここで紹介するアドベンチャーゲーム28選は，愛読者カードで人気の高いもの，
またTHE SOFTOUCH評価グループの推薦をもとに作成しました。

ソフト ＼ 内容	タイプ／入力方法／ストーリー	機種/メディア/価格/メーカー
パラレルワールド	ロールプレイング。英字入力：7つの並行世界にあるパラダイスを求めさまよい歩く。本誌「マシン語・魔神語～」でおなじみの長瀬敏之氏が制作。	X1/C/D(G-RAM要，BASIC) －T－ 3,800円Ⓔ
スターライトアドベンチャー PART Ⅰ	テキストアドベンチャー。Yes, No方式：俺は宇宙の旅行者。今日は8つの星の太陽系にやってきた！さあ何が起こるか？ 楽しみだぜ！	X1/C/D, MZ-2200/2000(G-RAM要) －T－ 3,500円Ⓒ
スターライトアドベンチャー PART Ⅱ	テキスト&グラフィック。Yes, No方式：宇宙空間に繰り広げられる，イマジネーションの世界。PARTⅠを超えた，グラフィック・ストーリーが君に迫る！	X1/C/D, MZ-2200/2000(G-RAM要) －T－ 3,800円Ⓒ
コスモクロス	リアルタイム&アドベンチャー：宇宙戦争の中，地球を救うために，出発した「セントコスモ号」のストーリー。総テキストメモリサイズ150Kバイト以上，10回を超える分割ロード。開発期間10カ月のクロスオーバーゲーム。アドベンチャーとしては，ちょっと毛色が違う。	X1/C/D(G-RAM要，BASIC) －T－ 4,800円 －3CF－ 5,800円Ⓧ
聖なる剣	テキストアドベンチャー。カナ入力：中世風の異世界を舞台としたファンタジック・アドベンチャー。	MZ-700(S-BASIC) －T－ 4,200円Ⓧ
ソフトハウス殺人事件	カナ入力：「あのLATX SOFTで殺人事件が！」ソフトハウスで起きた殺人事件に探偵クマダは捜査を開始した。XTALソフトの最新作。	MZ-2200/2000(G-RAM要 MZ-1Z002) －T－ 4,200円 4月末発売予定Ⓧ
名探偵登場	英字入力：会社をクビになった男が次に選んだ職業は，「クリスタル探偵事務所」の探偵だった。新米探偵の取り組む事件は，はたして成功するか。本誌3月号THE SOFTOUCHでも紹介したアドベンチャーです。	MZ-2200/2000(MZ-1Z002) －T－ 4,500円Ⓧ
ポイボスPart 1 －脱出－	ロールプレイング。英数字入力：長老ティタノアから宝玉を手渡され，ポイボスの再建を誓うジョーグは，ジルダ＝3収容所を脱走し仲間を集め，数々の困難に打ち勝って，クラーネ星を脱出せねばならない。行け！ ジョーグ，輝ける大地を求めて。	X1/C/D(G-RAM要，BASIC) MZ-700(Hu-BASIC) －T－ 3,800円Ⓩ
幽霊船	テキストアドベンチャー。カナ入力：幽霊船に隠された宝物を，怨霊のタタリを避けながら探す知的アドベンチャー。	X1/C/D, MZ-2200/2000/80B,MZ-700(S-BASIC) －T－ 3,500円㊐
ワンダーハウス	英字入力：不思議な家，ワンダーハウスで，ダイヤモンドを探し出せ。	X1/C/D. MZ-2200/2000/700/1200/K/C －T－ 3,500円㊅
南太平洋アドベンチャー	カナ入力：UCLA留学生チャン・ギステマラはミクロネシアの隠し財宝を求めて出発した。しかし途中，飛行艇が嵐におそわれ，チャンは南太平洋の海をさまようことになってしまった。	X1/C(G-RAM. dB-BASIC要) －T－ 4,500円ⓓ
スターアーサー伝説Ⅰ 惑星メフィウス	カナ入力：スターアーサーは伝説の剣を求めて，今，未知の惑星メフィウスへ降り立った。連作として発表されたスターアーサー伝説がX1シリーズについに登場。	X1/C(G-RAM要) －T－ 3巻 4,800円Ⓣ
首相の犯罪	カナ入力：その時日本中の目がひとりの男へ集中した。果たしてその時，首相は犯罪を犯したか？戦後未曽有の疑獄事件を，あなたは検察官として追及する。一時，新聞でも話題になった，疑獄アドベンチャー。	X1/C(G-RAM要) －T－ 4,000円㊈
デゼニランド	英字入力：埼玉・千葉の血の抗争が純金の〝三月磨臼〟を永遠の謎にしてしまった。あなたは〝三月磨臼〟を探し出せるか。 かつてない114画面のアドベンチャーランド。	X1/C/D(G-RAM要) －T－ 4,800円－3CF→5,800円Ⓗ
モンスターハウス	英字入力：ある日あなたの所へ手記が届いたが，なぜ今ごろこの手記が届いたのか？ 彼は2年も前に行方不明となっていたのだ……。 あなたは，疑問を解くために，彼の語っていた館へと出かけた。	MZ-2200/2000/80B(G-RAM1要) －T－ 4,500円Ⓕ

AUTOCRACY	英語入力：不思議な夢を見た朝，頭を冷やすために散歩をしようと思いドアを開け一歩外へ踏み出すとそこは別世界であった。アップル風本格的アドベンチャー。表示，入力ともに全部英語，画面数120枚の大作。	MZ-700（Hu-BASIC） －T－ 3,800円Ⓑ
連作タイムシークレット 第1話 ファラス星の危機	英字入力：タイムマシン研究所の君はタイムマシン試作号「メタフォトン号」に乗り，地球の歴史の中からダナーク人の攻撃法を見つけだし，ファラス星救出のため，時空間の旅に出ねばならない。THE SOFTOUCH紹介以来，人気のロールプレイング型アドベンチャー。	X1/C －T－ 4,000円。 MZ-700，MZ-1200/K/C －T－ 各3,300円㊙ X1，MZ-1200/K/Cは4月発売予定
不思議の森の アドベンチャー	カナ入力：登録語は100を超え，41画面４方向164の選択肢があります。ミステリアスな森の中に迷い込んだあなたは，みごと使命を達成できるか？ 本誌THE SOFTOUCH '83年12月号でも紹介したアドベンチャー。	MZ-700/1200/K/C －T－ 3,000円㊙
ペンギンビレッジ	記号入力：トリ山さんのバースデーパーティに招待され，野を越え山越え行く，ドリームアドベンチャー。おなじみアラレちゃんの世界です。	X1/C，MZ-2200/2000（G-RAM要，MZ-1Z002） －T－ 2,800円Ⓟ
宝ビルアドベンチャー I	英字入力：宝ビルの５Fにあるマイクロキャビンまで無事にたどり着けるか。 パートＩでは，エレベーターに乗るまで。	MZ-700/1200/80K/C －T－ 2,800円Ⓜ
宝ビルアドベンチャー II	英字入力：パートＩをクリアした人が楽しめるゲーム。 一部分リアルタイムを使用して，より難しくなっている。	MZ-700/1200/80K/C －T－ 2,800円Ⓜ
ドリームランド	英字入力：春の日ざしにうとうとと眠りこんだあなたは，いつのまにか夢の中をさまよいはじめた。そこはファンタジックな不思議な世界で，夢の出口を見つけなければ目ざめることができません。	MZ-80B，MZ-2200/2000（G-RAM1要）—5FD— 2枚組 各12,800円Ⓜ
不思議の国のアリス	テキストアドベンチャー。英字・カナ選択入力：ルイス・キャロル原作の「不思議の国のアリス」をアドベンチャー化したもの。テキスト・アドベンチャーだからこそ，あなたの想像力と勝負。	MZ-2200/2000/80B（G-RAM1要），X1（G-RAM要） －T－ 各3,500円 4月中旬発売予定Ⓜ
ミステリーハウス I	英字入力：電子小説といわれるアドベンチャーゲームのブームを築いた名作で，ロングセラーを続けている。夜もふけ自分だけの時間を持てたとき，ゆっくりと時間をかけて解いてください。	MZ-80B，MZ-2200/2000（G-RAM1要） －T－ 各5,200円 －5FD－ 各7,800円Ⓜ （X1用も近日発売予定）
ミステリーハウスII Vol Ⅰ．Vol II	大ヒットミステリーハウス第２段はVol Ⅰ，Vol IIの連作。ラインナップがそろっていっそうファンをとりこにする。	MZ-80B，MZ-2200/2000（G-RAM1要） －T－ Vol Ⅰ，II共各3,800円 －5FD－ Vol Ⅰ＋IIで各9,800円Ⓜ
黄金の墓	カナ入力：一枚の手紙があなたをロマンと冒険の旅へとかりたてた。 悪漢どもとの戦い，美少女との出会い，猛獣の襲撃，そしてクフ王の秘密とはなにか。砂漠に繰り広げられる壮大なアドベンチャードラマ。	X1/C/D（G-RAM要）。 －T－ 1巻4,800円 －3CD－ 6,200円㉘
ムー大陸の謎	英字入力：幻の「ムー大陸」の秘密を解くカギは？ 巨大な石像モアイ，未解読のロンゴ・ロンゴ文字，奇怪な偶像モアイ・カワ・カワなど，あなたに挑みかかる謎，また謎！ 頼れるのは知力と推理だけだ。	X1/C/D（G-RAM要）。 －T－ 4,800円 －3CD－ 6,200円㉘
ピラミッドの謎	英字入力：行方不明となった探検隊を探して，あなたはピラミッドの中へと入っていった。複雑な迷路や数々の恐しいワナがあなたを待ちうけているミステリアスなアドベンチャー。	X1/C/D（G-RAM要） －T－ 4,800円 －5FD－ 5,800円 －3CD－ 6,200円㉘

注.）表内のソフトは，メーカー名の五十音順，同メーカーソフトについては五十音順になっております。

○表中の記号について

T：テープ版　5FD：5インチフロッピーディスク版　3CD：3インチコンパクトフロッピー版

問い合わせ先

Ⓔ(株)エニックス 〒160 新宿区西新宿7-1-8 ☎03-367-1911 ／ ©キャリーラボ 〒862 熊本市大江6-25-25金子ビル1F ☎096-363-0211 ／ Ⓧクリスタルソフト(株) 〒533 大阪市東淀川区豊新3-8-11いほりビル2F ☎06-326-8150 ／ ㉗大名マイコン学院（ZAT SOFT） 〒810 福岡市中央区大名2-6-1福岡国際ビル6F ☎092-715-8526 ／ ㊊(株)新紀元社 〒160 新宿区新宿4-1-9新宿ユースビル8F ☎03-350-0848 ／ ㊎タスクフォーツ高知 〒780 高知市朝倉丙434-14 ☎0888-40-1207 ／ ⓓデービーソフト 〒060 札幌市中央区北3条西2丁目カミヤマビル4F ☎011-251-7462 ／ Ⓣ(株)ティーアンドイーソフト（T＆E SOFT） 〒465 名古屋市名東区藤が丘141藤が丘ビル4F ☎052-773-7770 ／ ㊏東映ビデオ(株)（FILCOM） 〒104 中央区銀座3-15-10八重洲興業銀座ビル7F ☎03-545-4511 ／ Ⓗハドソン東京（HUDSON SOFT） 〒102 千代田区麹町4-7-5麹町ロイヤルビル2F ☎03-234-4996 ／ Ⓑ(株)計測技研（BASIC HOUSE） 〒320 宇都宮市桜3丁目2-17 ☎0286-33-1994 ／ ㊌(株)バスカルII（BOND SOFT） 〒430 浜松市常盤町140-2 ☎0534-53-6186 ／ Ⓟ(株)ポニーPONYCA開発室 〒102 千代田区九段北4-1-3日本ビルディング3F ☎03-265-6377 ／ Ⓜ(株)マイクロキャビン 〒510 四日市市鵜の森1-2-15メゾンヴァンベール２F ☎0593-51-6482 ／ ㉙ストラットフォード・コンピューターセンター(株)ホビー事業部(MagicalZoo) 〒336 浦和市南浦和2-36-15 ☎0488-85-5222

これがプロのテクニックだ

——不思議の国のアリス——

㈱マイクロキャビン　多湖　明雅

はじめに

「ミステリーハウス」を先駆とするアドベンチャーゲームも，最近ではかなり数多く出てきています。基本のテキストアドベンチャーものから画面数が100を超えるものまで，まだまだ可能性をもつジャンルですが，今回，Oh！MZの読者の皆さんのために，あの「ミステリーハウス」を作り上げたマイクロキャビンより，アドベンチャーゲーム作りのKnow-Howを公開していただきました。今回のテキストとなるゲームは，PC用で発売されている，テキストアドベンチャー「不思議の国のアリス」で，MZ-80B/2200/2000はもちろん，MZ-5500でも，「システムソフトウェアコンバータ」(マイコンショップBASIC HOUSE)を使えばX1でも，またS-BASICを使って，一部修正すればMZ-700でも作動します。
まずは扉をたたいてみましょう。

アドベンチャーゲームの構造1

まず，このプログラムの特徴を説明しながら，アドベンチャーゲームの構造について説明していきます。

このアリスのプログラム構造は，基本的にはシーン・ブロックによって形成されています。たとえば，今アリスが駅にいるとしましょう。この状態でアリスにできることは，キップを買うこと，電車に乗ること，駅から出ることです。ここで，この仮想場面をプログラムの形にしてみましょう。

```
10 PRINT〝アナタ ハ イマ エキ
   ニ イマス。〟
20 INPUT A$
30 IF A$＝〝キップ ヲ カウ〟TH
   EN モチモノ＝モチモノ＋キップ:
   GOTO 20
40 IF A$＝〝デンシャ ニ ノル〟T
   HEN GOTO ＊ 電車の中の仮想シーン
50 IF A$＝〝エキ ヲ デル〟THEN
   GOTO＊ 駅前通りの仮想シーン
60 PRINT〝ソノヨウ ナ コウドウ
   ハ デキマセン。〟
70 GOTO 20
```

とまあ，ざっとこんなものでしょうか。実際はもっと複雑なものになりますが，これがいわゆるシーン・ブロックというものです。ちなみに，あの「ミステリーハウス」は全面的にこの方法を用いています。

ところで，上記のプログラム中に，「モチモノ」という変数が出てきましたが，アドベンチャーゲームのプログラム中では，これらの変数が実に重要な働きをします。この種の変数は，ほかにも多種多様にわたって使われます。アリスがドアの鍵を持っているかいないか，ドアが開いているかいないか，時計が動いているかいないか，などなど，およそゲーム中に出てくるものはすべて0か1などの数字によって，それぞれの変数に収められることになります。いわばこの仮想世界は，すべて個々の数字によって構成されているといってもいいのです。どうです？　イメージが浮かびましたか？

このことはゲームに重大な影響を与えます。たとえば，先ほどのプログラムの続きで，アリスが電車に乗ったとします。そして四日市で降りたとします。そこで改札を通るとき，プログラムは先ほどのモチモノという変数を調べます。キップを持っているかいないかは，そこに0か1という形で明確に表われています。駅員の代わりをするプログラムは，もしアリスがキップを持っていれば「ありがとうございました」と言って通してくれ，もし持っていなかったら，問答の末，近鉄全線の3倍の料金を請求され，お金を持っていなかったアリスは，一生近鉄でタダ働きをさせられる，といったふうです。

もうだいぶ理解されたと思いますが，一応これをプログラムに落としてみましょう。

```
10 IF A$＝〝エキ ヲ デル〟AND
   モチモノの中にキップがある THE
   N PRINT〝アリガトウゴザイマシ
   タ。〟:GOTO＊駅の外の仮想シーン
20 IF A$＝〝エキ ヲ デル〟AND
   モチモノの中にキップがない TH
   EN PRINT〝ナントユーヤツダ，
   オマエ ノ ヨウナヤツハ，イッシ
   ョウウツベセン ノ エキイントシ
   テタダバタラキサセテヤル。ガミガ
   ミ……〟:END
```

だいたいこういう具合になります。

さて，今までお話ししてきたプログラムの構成には，大きな欠点があります。たとえば，仮にアリスの舞台設定をドアだらけの迷路みたいな家だとします。すると，ドアの数だけの変数とドアの数×2（同じドアでも別の部屋から見ると別の処理ルーチンとなる）の処理ルーチンが必要です。これは，ドアの数が多くなればなるほどメモリを食うわけです。TAKEやTHROWなども，あらゆる可能性を考えて，プログラムに組み込まなくてはいけません。

では，なぜこの方法を採用したかというと，作るのが簡単だからです。誰でも簡単にアドベンチャーゲームが作れるということです。前にあげたプログラム例からも，この方法だと頭に描いたイメージがダイレクトにプログラムに落とせるということが

わかると思います。そのほかにも，まったく処理系の違ったプログラム，たとえばリアルタイムゲームなども簡単に組み込むことができます。また，複数で並行してプログラム作りをする場合，製作・結合，そのどちらも簡単にできます。

ご覧になったように，個々のシーン・ブロックはまったく独立したものとして存在し，使用変数さえきちんと統一しておけば，あとでブロックを組み合わせれば完成するというわけです。この方法は開発期間が限られているとか，複数で共同開発する場合に大きな効果があり，これらメリットのため，今日でも比較的多用されているようです。

アドベンチャーゲームの構造2

次に，今までお話ししてきたのとは別の，アドベンチャーゲームのための処理構造をお話ししましょう。

これは，主に入力された動詞別に処理を行なう方法です。動詞の種類によって分類された先で，さらに名詞によって判断され，必要なものはさらにシーン別に判断されます。

さらにこの方法には，ほとんどの場合において，データテーブルによる処理方法も取り入れられています。

この処理方法は，パラメータ（入出力語群や出力メッセージ群など）をデータとして扱い，フラグ（文字どおり「旗」という意味で，代入されている数字によって何らかの状態を表わす変数）群を配列に収めて一括処理ができるようにし，アドベンチャーゲーム中の判断部分（IF文）を最低限におさえるという方法です。

プログラムはまず，入力された語句を，持っているデータと順次比較検討し，相当する単語に対してフラグを操作し，出力データの中からその時点の状況に見合うものを選び出して表示します。また，これらのデータは一定の規定に従って並べられており（テーブル化），プログラムの最後または先頭にまとめて置きます。

ここでひとつ，この方法におけるプログラムの流れを考えてみましょう。

シーンナンバー（その場面につけられた番号）が3の場所で，〝OPEN〟と入力された場合，プログラムはまず〝OPEN〟という単語を識別して〝OPEN〟の処理ルーチンへ飛びます。そしてその中で〝DOOR〟を識別し，そのシーンのドアのフラグを立て（1を書き込む），ドアが開いたことを示します。

ここで注目してほしいのは，ドアを表わす変数に，添字をシーンナンバーにした配列変数を使うと，ひとつの処理ルーチンで全部のドアが操作できるということです。この方法は，多くの部屋で同じことを繰り返して行なうような，たとえば家の中の宝探しなどのゲームには，メモリの節約になり大いに有効です。ちなみに「幽霊船」，「鍵穴殺人事件」などが，この方法を採用しています。

ただこの方法の欠点は，前もってアドベンチャーの舞台設定をしっかり決めておく必要があります。シーンの変更や削除，追加などたいへんな労力を必要とし，わけのわからないバグなども出やすく，精神衛生上好ましくありません。ですから，あまり初心者にはおすすめできる方法ではありません。

両方を兼ね備えた新しい方法を

さてここで，本プログラムです。「不思議の国のアリス」のストーリーを用いて，実際にプログラミングをするにあたり，今まで述べてきた2つの方法を兼ね備えた新しい方法をということで考えてみました。どういうことかというと，どこでも使うことのできるような一般的な動詞（LOOK, TAKE, THROW, EAT, INVENTORY, SAVE, LOAD）は動詞別に分岐させる方法で処理させ，特殊な，または変則的な動詞は各シーン別の処理ルーチンへ飛ばす，というような方法です。単語群やメッセージ群はできる限りデータ化し，シーン中に出てくる物は，すべて圧縮，データ化してあります。

また，データの扱いですが，名詞の中でDOORやHOUSEなど，同じ単語が複数個使われているのは，1つひとつを区別して表わすためです。たとえば，白ウサギと三月ウサギはともにRABBITですが，まったく別のキャラクタとして扱います。

これに対して動詞は場面ごとに区別されていません。すべての場面に共通の一単語で動作を表わしています。

移動可能な名詞と移動不可能な名詞も区別して配列に格納しています。

また，このプログラム特有なものとして，体の大きさを表わすフラグがあります。したがって，これに類する一種の状態（眠っている，力が強くなっている，魔法をかけられているなど）についても考慮しなければなりません。しかし，ごく短期間の状態なら，個別に処理したほうが有利なこともあります。

データ圧縮についてはいろいろな方法がありますが，ここでは2進数の各ケタをフラグとして取り出し，必要でないときは10進数にしてしまっておくという方法をとっています。暗号化を考えるなら，素数をいくつかかけていき，特定の素数で割って，割り切れるか否かを確かめるのもよいでしょう。

プログラムの解説

それでは以上のことを念頭におきながら「不思議の国のアリス」のプログラムを解説したいと思います。

●初期設定　8620～9740行

イニシャライズ後，メインルーチンへ飛びます。ここで各データの変数への格納，および各変数の初期設定を行ないます。

持ち物，各シーンにある物は，それぞれひとつの変数に圧縮されて格納されています。これは主にメモリ効率を上げるためです。マシン語化への布石ともなるように2進数で表現してあります。

最後のほうに2のべき乗がわざわざ配列中に入れてありますが，これはひとえにPC-8001のせいです。2のべき乗は，10進→2進変換のところで使うのですが，PC-8001にこの計算をやらせると誤差が大きくて実用になりません（家の中に突然木がはえたりするのです）。僕はプログラムのバグかと思い必死で悩みましたが，誤差とわかり，腹立ちまぎれに，使いそうな2のべき乗はみんな配列に放り込んだというわけです。MZ用にも，スピードが上がるという利点があるためそのまま残しています。

●メインルーチン

コマンド入力，動詞・名詞分割，動詞判断，名詞判断，コマンド・シーンへの分岐，メッセージ部分からなっています。

動詞・名詞分割は英語用，日本語用別に作ってあります。日本語用のほうは，助詞が入ってもいいように処理しているのでち

表1　動詞

LOOK……みる	OPEN……あける
TAKE……とる	SAY……いう
THROW……なげる	TALK……はなす
EAT……たべる	UNLOCK……かぎをあける
INVENTORY……もちもの	WEAR……はめる
GO……いく	SAVE……セーブ
BACK……もどる	LOAD……ロード
ASK……きく	CLIMB……のぼる
DRINK……のむ	OPEN……(かさを)さす
ENTER……はいる	JUMP……とびおりる
FAN……あおぐ	CLOSE……とじる
GIVE……あげる	DOWN……おりる
MOVE……うごかす	FOLLOW……ついていく

表2　名詞（移動不可能）

BUTLER……しつじ	STORE……みせ
CAT……ねこ	TABLE……テーブル
CATAPILLAR……いもむし	DOOR……ドア
CHIMNEY……えんとつ	DOOR……ドア
CURTAIN……カーテン	DOOR……ドア
DOG……いぬ	HOUSE……いえ
DOOR……ドア	HOUSE……いえ
DRESSER……きょうだい	RABBIT……うさぎ
HOUSE……いえ	SMALL DOOR……ちいさいドア
KING……おうさま	TREE……き
MARCHIONESS……こうしゃくふじん	WALL……かべ
QUEEN……じょおう	TREE……き
RABBIT……うさぎ	TAKO……たこ
SMALL DOOR……ちいさいドア	

表3　名詞（移動可能）

BOTTLE……びん	
FAN……うちわ	
GLOVE……てぶくろ	
HAT……ぼうし	
HATCHET……おの	
KEY……かぎ	
MUSHROOM……きのこ	
STICK……ぼう	
PARASOL……パラソル	
ANSWER……こたえ	
HELLO……こんにちは	
JUICE……ジュース	
MYSELF……じぶん	
OUT……オモテ	
EYE……め	

よっと複雑です。実際は助詞まで判断していないのですが，ちゃんと調べるのもいいでしょう。

動詞・名詞判断のところで，入力したものと一致したときのデータの番号を使い，これを組み合わせて(上位2桁…動詞番号，下位2桁…名詞番号)，入力したコマンドの代わりにしています。これによって，メモリの節約（IF A$＝〝OPEN DOOR〟よりもIF A＝1032のほうが数バイト少ない）や，一見しただけでは，答えがわからないプログラムを作ることができるという利点が生じます。

その他，THROWやTAKEの特殊処理も行なっています（内容はひみつ♥）。

●一般動詞の判断

いよいよ各動詞の判断です。まずLOOKサブルーチンです。ここで注意してほしいのは，文字型アドベンチャーゲームでは，場面が進むごとに，情景描写を行なうということです。つまり場面が進む時点で，自動的にLOOKが入力されるということです。さらに，絵が出るタイプのものに比べて，細かい説明のついた文を出さなければなりません。よって，このLOOKサブルーチンは，汎用性に富んでいなくてはなりません。

LOOKの対象となる名詞も情景も，すべて変数に代入されていますから，これをシーンナンバーから算出して表示します。入力のとき，名詞を指定したときの場合分けも必要です。また，さらに持ち物の描写を詳しく行ないたい，という人の参考例として，ここでは持ち物に対してLOOKを行なった場合「＊carring＊」というメッセージも同時に出すようにしてみました。

さらに，LOOKして初めて出現する物の処理も行なっています（たとえば，TABLEをLOOKしたら，その上にKEYが乗っているのを見つけた，という類の処理です）。

次は，TAKEサブルーチンです。ここに至って，ようやく固定物と可動物の分離の恩恵にあずかれます。可動物の場合，それを持ち物の中に取り入れます。

「不思議の国のアリス」では「体の大きさ」というパラメータが存在するので，ここではその処理（体が小さいと，手がとどかない！）をしています。

次にTHROWです。TAKEと逆のことをやっています。コマンド別処理をした利点が一番よく表われています。

EATルーチンです。ここは主にきのこを食べたときの処理（詳しくは，な・い・しょ♥）をやっていますが，シーン中に出てくる物を食べたときも，一応処理はしています。

INVENTORYです。「持ち物を調べる」という意味で使っています。持ち物の変数を調べ，ビットが立っている名詞を表示していきます。

最後にSAVE・LOADルーチンですが，これはあとでつけたしたコマンドなので，プログラムの最後のほうに置いてあります。ここでは，変わる可能性のある変数をすべて，データとして外部記憶装置に書き込みます。それをもとの変数にLOADしなおすことによって，前のゲームの状態を再現します。

以上がメインルーチンと一般動詞処理の説明ですが，結構多くのテクニックを盛り込んでありますので，わかりにくい個所が多いと思います。しかし，プログラミングのネックとなるのは，フラグの設定と処理の順序などですから，自分がわかりやすい構成でプログラムを組み立ててみてください。そして行き詰まってしまったら，この「不思議の国のアリス」の中に，うまい解決策がないかと，探してみてください。きっとよい参考になると思います。

次は，各シーン・ブロックの解説です。シーンナンバーはプログラム中のリマーク文に書いてあるので参照してください。

●Scene 1

まず最初に，この場面で有効なコマンドを判別しています。有効でないコマンドに対しても，特別なメッセージを出力したいときにはこの部分で処理してから，再度，入力待ちに戻します。

有効なコマンドに対しては，それに応じた処理を行ないます。このときも主役となるのはフラグ操作です。ある命令が入力されたとき，これが成立するか否かをフラグによって判断します。たとえばドアを開け

表4　変数表

TH（0, X）	シーンXにおいて移動しない登場物
TH（1, X）	シーンXにおいて移動する登場物
H（0）	持ち物
H（1）	体の大きさ（0…小さい，1…大きい）
CO	各シーン・ブロックに受け渡す動詞，名詞の通し番号（動詞ナンバー×100＋名詞ナンバー）
VR$（X）	動詞単語群
NA$（0か1, X）	名詞単語群（0…移動しないもの，1…移動するもの）
FM$（X）	メッセージ文章群
MA$（X）	シーンナンバーXにおいて，そこにあるものと同じ名前のものがすでに使われているとき，それを無視するようにするための変数。無視する変数は2桁ずつ順に入っている。

るときなら，カギはかかっていないか，体が小さいときは，ドアのノブに手が届かないなどの条件はどうか，すでに開いていないか，などを1つひとつ検討します。

この「入力待ちに戻す」処理は，非常に頻繁に行なわれるため，各場面はそれぞれサブルーチンとして取り扱われています。これなら，RETURN文ひとつでメインルーチンのループに戻すことができます。しかし，このときは，GOSUB文との対応をしっかりと覚えておいてください。そうしないと，戻るべき場所を見失ってしまいます。プログラミング上達のカギは，ここらあたりの注意力にあるのです。

●Scene 2

何度も同じコマンドが入力されたとき，常に変わらぬ表示をするのは，わかりやすい反面，面白さに欠けます。というわけで，ここでは，何度もウサギに話しかけると，「何度も話しかけないでくれ……」と答えるように工夫をしてみました。もちろん，「すでにウサギと話したか否か」のフラグを用いていることは言うまでもありません。

●Scene 3

ゲームが途中で終わってしまう例です。このプログラムでは，途中で終わってしまう理由を説明したのち，END文で終わってしまいますが，再度ゲームを行なうか否かを尋ねるようにしてもよいでしょう。このとき，変数の初期化に気をつけてください。初期化しないと，とんでもないフラグが立ったままになる恐れがあります。CLEAR文を用いると効果的でしょう。

●Scene 4～12, 14, 15, 18

特に書くことはないでしょう。比較的簡単な部分なので，フラグの動きを追いながらプログラムを読んでみてください。

●Scene 13

迷路のような森をぬけるときの方向の入力をここで独立に処理しています。

部分的な処理による入力なので，方向のほかのコマンドは受け付けなくなってしまいますが，工夫次第では解決できると思います。そのためにはコマンド入力部分がサブルーチン化されていなければなりません。このサブルーチンをコールすることによって，どんな場面においても入力の判断を行なうことができるのです。ただし，くれぐれもGOSUB RETURNの対応には気をつけてください。ヘタすると，自分自身をコールしてしまった（再帰的呼び出しはBASICでは許されていません）というデバッグの困難なバグを生じることになります。

ただ「不思議の国のアリス」では，コマンド入力それ自体がメインルーチンとなり，各場面処理などをサブルーチンコールする，という構成のため入力の判断を行なうのは困難です。

●Scene 16, 17, 19

このあたり，体の大きさの変化に気をつけて読んでください。

体の大きさ，というような即物的でないものは，持ち物のリストをとっても表示されませんから，表現を工夫して，プレイヤーにうまく悟らせる必要があります。

●Scene 20

いよいよクライマックスです。このプログラムでは，終わりに近づくほど，出力メッセージが長くなっていますが，「ここまで作ればできたも同然」というわけで，思わずいろいろと書きたくなるものなのです。

たくさんの表示を一度に行なうときに，ゆっくり，徐々に表示していくという方法をとっています。

●最終場面処理

アドベンチャーゲームには，必ず目的とするものがあり，エンディングまで到達したということは，当然その目的を果たしたということになります。そこで，エンディングメッセージは，祝福の言葉となるのが普通です。しかし，あまりにあっけなく，「CONGRATULATION」では，せっかくここまできた努力も報いられません。美辞麗句を並べたてるのもよいのですが，あっ！と驚く意外な結末というのも楽しくて良いかも知れません。

これで僕の解説は終わりです。みなさんも一度アドベンチャーを作って，友人などに遊ばせてやってください。友人の苦しむ顔を見ると，やみつきになりますよ!!

パソコンドリーム
第1回
アドベンチャーゲームコンテスト

最優秀アドベンチャースピリッツ賞
100万円1名
ほか豪華な賞金，賞品を多数用意してあります。

未発表のオリジナルアドベンチャーゲームまたはロールプレイングゲームで，ストーリー，キャラクタなどで他の著作物の著作権を侵さない作品を募集します。また，ストーリー部門のみ，シナリオだけでも受け付けます。

◎応募要領
機種は特に問いません。言語はBASIC，マシン語，BASICコンパイラで，カセット，ディスケットどちらでも構いません。詳しくは
〒510 三重県四日市市鵜の森1－2－15
メゾンヴァンベール2F
☎0593-51-6482
(株)マイクロキャビン
にお問い合わせください。

リスト

```
1000 REM"** START
1020 GOSUB 8620
1030 GOSUB 1970
1040 GOTO 1220
1050 REM"** 10ｼﾝｰ2ｼﾝ
1060 A=H(0)
1070 A$=""
1080 FOR B=26 TO 0 STEP -1
1090 IF A>=TA(B) THEN A=A-TA(B):A$=A$+"1":GOTO 1110
1100 A$=A$+"0"
1110 NEXT
1120 RETURN
1130 REM"** LETTER
1140 B$=A$
1150 A$=""
1160 FOR I=1 TO LEN(B$)
1170 A=ASC(MID$(B$,I,1))
1180 IF (96<A)*(A<123) THEN A=A-32
1190 A$=A$+CHR$(A)
1200 NEXT
1210 RETURN
1220 REM"*** MAIN
1230 PRINT
1240 PRINT "[ ENTER COMMAND ]"
1250 GOSUB 10300
1260 PRINT
1270 IF A$="" THEN 1220
1280 GOSUB 1140
1290 REM"** ﾄﾞｳｼ,ﾒｲｼ
1300 A=1
1310 IF LU$="J" THEN 1380
1320 IF MID$(A$,A,1)=" " THEN A=A-1:GOTO 1350
1330 IF A<>LEN(A$) THEN A=A+1:GOTO 1320
1340 I2$="":GOTO 1360
1350 I2$=MID$(A$,A+2,LEN(A$)-A-1)
1360 I1$=LEFT$(A$,A)
1370 GOTO 1480
1380 IF MID$(A$,A,1)=" " THEN B=A+1:A=A-1:GOTO 1430
1390 IF A<>LEN(A$) THEN A=A+1:GOTO 1380
1400 I1$=A$
1410 I2$=""
1420 GOTO 1480
1430 IF MID$(A$,B,1)=" " THEN B=B+1:GOTO 1460
1440 IF B<>LEN(A$) THEN B=B+1:GOTO 1430
1450 B=A+2
1460 I2$=LEFT$(A$,A)
1470 I1$=MID$(A$,B,LEN(A$)-B+1)
1480 C1=1
1490 IF I1$=VR$(C1) THEN 1530
1500 C1=C1+1
1510 IF C1<27 THEN 1490
1520 PRINT "ｿﾚﾊ ﾃﾞｷﾏｾﾝ。":GOTO 1220
1530 FOR I=0 TO 3
1540 CH(I)=VAL(MID$(MA$(NO),2*I+1,2))
1550 IF CH(I)=0 THEN CH(I)=99
1560 NEXT
1570 A=0
1580 FL=0
1590 BT=0
1600 IF BT=CH(A) THEN A=A+1:GOTO 1620
1610 IF I2$=NA$(0,BT) THEN 1720
1620 BT=BT+1
1630 IF BT<27 THEN 1600
1640 FL=1
1650 BT=0
1660 IF I2$=NA$(1,BT) THEN 1720
1670 BT=BT+1
1680 IF BT<15 THEN 1660
1690 IF I2$<>"" THEN PRINT I2$;" ??":GOTO 1220
1700 CO=C1
1710 GOTO 1740
1720 CO=C1*100+BT+FL*27
1730 IF (FL=1)*(BT>8) THEN FL=3:GOTO 1810
1740 IF C1>5 THEN 1810
1750 IF H(15)<>0 THEN 1790
1760 IF C1<>1 THEN 1790
1770 PRINT "ﾅﾝﾆﾓ ﾐｴﾅｲ。"
1780 GOTO 1220
1790 ON C1 GOSUB 1970,2370,2510,2620,2980
1800 IF ((NO<>5)+(CO<>334))*((NO<>17)+(CO<>230)) THEN 1220
1810 IF C1=19 THEN 9750
1820 IF C1=20 THEN 9970
1830 OL=NO
1840 IF NO>10 THEN 1870
1850 ON NO GOSUB 3100,3600,3860,4440,4740,4930,5120,5350,5460,5860
1860 GOTO 1880
1870 ON NO-10 GOSUB 6050,6260,6480,6720,6820,7070,7180,7500,7740,8040
1880 IF OL<>NO THEN RESTORE 1890:GOSUB 10550:CO=1:GOSUB 1970
1890 DATA "/","ﾊﾞｼｮ ｶﾞ ｶﾜﾘﾏｼﾀ。/","*"
1900 GOTO 1220
1910 PRINT "OK.":RETURN
1920 USR($0F14):PRINT "ﾀﾞﾒﾃﾞｽ。":RETURN
1930 USR($0F14):PRINT "ｽﾃﾞﾆ ｱｲﾃ ｲﾏｽ。":RETURN
1940 USR($0F14):PRINT "ｶｷﾞ ﾊ ｱｲﾃｲﾏｽ。":RETURN
1950 USR($0F14):PRINT "ｶｷﾞ ｶﾞ ｶｶｯﾃｲﾏｽ。":RETURN
```

MZ-700(S-BASIC)で打ち込む方の変更点

リスト中すべてのUSR($0F14)を
USR($3E)に変更
以下次の行を変更してください。

1130 ⟩ 必要ありません 1210

1250 INPUT ">␣";A$

1280 必要ありません

```
1960 USR(&HF14):PRINT "カギ ガ アリマセン。":RETURN
1970 REM'** LOOK
1980 IF CO>1 THEN 2140
1990 PRINT:PRINT " ココハ ";NO$(NO);" デ";
2000 E=0
2010 FOR D=0 TO 1
2020 A=TH(D,NO)
2030 GOSUB 1070
2040 FOR I=1 TO 27
2050 IF MID$(A$,I,1)="0" THEN 2090
2060 E=E+1
2070 IF E>1 THEN PRINT " ト";
2080 PRINT " ";NA$(D,27-I);
2090 NEXT
2100 NEXT
2110 IF E>0 THEN PRINT " ガ ミエマス。":RETURN
2120 PRINT "ス。"
2130 RETURN
2140 REM'* LOOK
2150 IF FL=0 THEN 2180
2160 GOSUB 1060
2170 IF MID$(A$,27-BT,1)="1" THEN 2220
2180 A=TH(FL,NO)
2190 GOSUB 1070
2200 IF MID$(A$,27-BT,1)="1" THEN 2230
2210 GOTO 1920
2220 PRINT "*** carrying ***"
2230 B=VAL(LEFT$(TR$(FL,BT),2))
2240 PRINT FM$(B);" ";NA$(FL,BT);" デス。"
2250 A=0
2260 A$=MID$(TR$(FL,BT),3,1)
2270 IF A$="*" THEN 2350
2280 A=1
2290 F=VAL(A$)
2300 B=VAL(MID$(TR$(FL,BT),4,2))
2310 PRINT " ";NA$(F,B);" ト";
2320 TH(F,NO)=TH(F,NO)+TA(B)
2330 TR$(FL,BT)=LEFT$(TR$(FL,BT),2)+MID$(TR$(FL,BT),6,LEN(TR$(FL,BT))-5)
2340 GOTO 2260
2350 IF A=1 THEN PRINT CHR$(29);"ガ アリマシタ。"
2360 RETURN
2370 REM'** TAKE
2380 IF (NO=1)*(CO=232)*(H(1)=0)*(H(20)=0) THEN 1920
2390 IF (NO=3)*(CO=234)*(H(1)=0)*(H(21)=0) THEN 1920
2400 IF FL=0 THEN USR(&HF14):PRINT "ソレ ハ モテマセン!":RETURN
2410 A=TH(1,NO)
2420 GOSUB 1070
2430 IF MID$(A$,27-BT,1)="0" THEN 1920
2440 GOSUB 1910
2450 TH(1,NO)=TH(1,NO)-TA(BT)
2460 H(0)=H(0)+TA(BT)
2470 IF (NO=1)*(CO=232) THEN H(20)=1
2480 IF (NO=3)*(CO=234) THEN H(21)=1
2490 IF (CO=233)*(H(19)=5) THEN H(19)=4
2500 RETURN
2510 REM'** THROW
2520 IF FL=0 THEN 2600
2530 GOSUB 1060
2540 IF MID$(A$,27-BT,1)="0" THEN 2600
2550 GOSUB 1910
2560 H(0)=H(0)-TA(BT)
2570 TH(1,NO)=TH(1,NO)+TA(BT)
2580 IF CO=329 THEN H(13)=0
2590 RETURN
2600 PRINT "モッテ イマセン。"
2610 RETURN
2620 REM'** EAT
2630 IF FL=0 THEN 2680
2640 GOSUB 1060
2650 IF MID$(A$,27-BT,1)="0" THEN 2680
2660 IF BT=6 THEN 2720
2670 GOTO 2820
2680 A=TH(FL,NO)
2690 GOSUB 1070
2700 IF MID$(A$,27-BT,1)="0" THEN 1920
2710 GOTO 2820
2720 IF H(19)=5 THEN 2600
2730 IF H(19)=0 THEN PRINT "モウ ゼンブ タベテシマイマシタ。":RETURN
2740 H(19)=H(19)-1
2750 IF H(19)=0 THEN H(0)=H(0)-64
2760 IF H(10)=0 THEN PRINT "コレハ ウマイヨ。":RETURN
2770 INPUT "ジュモンハ ";A$
2780 IF A$="ヒヒルマ" THEN 2840
2790 IF A$="ハハレホ" THEN 2880
2800 PRINT "オイシイ。"
2810 RETURN
2820 PRINT "マズイ..."
2830 RETURN
2840 REM'*
2850 IF H(1)=0 THEN GOSUB 1910:H(1)=1:RETURN
2860 PRINT "モウ オオキクナッテイマス。"
2870 RETURN
2880 REM'*
2890 IF H(1)=1 THEN GOSUB 1910:H(1)=0:RETURN
2900 RESTORE 2910:GOSUB 10550
```

```
2910 DATA "「オーイ! タコワン。 アリスチャン ガ゛ トツゼ゛ン イナクナッテ シマッタゾ゛ ?」/"
2920 DATA "「チカ゛ウンタ゛ CPUワン。 アリスチャンハ チイサクナリスギ゛テ"
2930 DATA "キミノメニ ミエナクナッタ タ゛ケナンタ゛!」/"
2940 DATA "「マルテ゛ ミクロ ケッシタイ タ゛ナー... /"
2950 DATA "シカシ ミエナキャ ケ゛ーム ハ ツツ゛クラレナイナァ....」/"
2960 DATA "ト イウワケテ゛  サヨーナラーァァァァ....",*"
2970 GOTO 10660
2980 REM'*** INVENTORY
2990 IF H(0)=0 THEN PRINT "ナニモ モッテイマセン。":RETURN
3000 GOSUB 1060
3010 PRINT "アナタ ハ";
3020 E=0
3030 FOR I=16 TO 27
3040 IF MID$(A$,I,1)="0" THEN 3070
3050 E=E+1:IF E>1 THEN PRINT " ト";
3060 PRINT " ";NA$(1,27-I);
3070 NEXT
3080 PRINT " ヲ モッテ イマス。"
3090 RETURN
3100 REM'**** SCENE 1
3110 IF CO=7 THEN PRINT "アトニハ ヒキカエセマセン。":RETURN
3120 IF CO=938 THEN 3210
3130 IF CO=1013 THEN 3310
3140 IF (CO=1304)+(CO=1404) THEN 3340
3150 IF CO=1406 THEN 1960
3160 IF CO=1413 THEN 3410
3170 IF CO=1706 THEN 3450
3180 IF CO=1713 THEN 3500
3190 GOTO 1920
3200 REM'*
3210 GOSUB 1060
3220 IF MID$(A$,27,1)="0" THEN 1920
3230 GOSUB 1910
3240 RESTORE 3250:GOSUB 10550
3250 DATA "キャ------..../"
3260 DATA "アナタ ハ マイクロ テ゛ィスク ク゛ライ ノ オオキサ ニ ナッテシマイ "
3270 DATA "シ゛ュース ノ ハイッタ ビ゛ン モ ト゛コカ ニ トンテ゛イッテ シマイマシタ。/",*"
3280 H(1)=0
3290 H(0)=H(0)-1
3300 RETURN
3310 REM'*
3320 IF (H(2)=3)*(H(1)=0) THEN NO=2:RETURN
3330 GOTO 1920
3340 REM'*
3350 IF H(2)<>0 THEN 1930
3360 GOSUB 1910
3370 H(2)=1
3380 PRINT "カーテン ノ カケ゛ニ ";NA$(0,22);" カ゛ アリマシタ。"
3390 TH(0,1)=TH(0,1)+8192
3400 RETURN
3410 REM'*
3420 ON H(2)+1 GOTO 1920,1950,3430,1930
3430 H(2)=3
3440 GOTO 1910
3450 REM'*
3460 GOSUB 1060
3470 IF MID$(A$,22,1)="0" THEN 1960
3480 PRINT "カギ゛ カ゛ チカ゛ウヨウテ゛ス。"
3490 RETURN
3500 REM'*
3510 ON H(2)+1 GOTO 1920,3520,1940,1930
3520 IF H(2)=2 THEN 1940
3530 GOSUB 1060
3540 IF MID$(A$,22,1)="0" THEN 1960
3550 GOSUB 1910
3560 H(2)=2
3570 PRINT "アイタトタン ニ カギ゛ カ゛ キエテシマイマシタ。"
3580 H(0)=H(0)-32
3590 RETURN
3600 REM'**** SCENE 2
3610 IF CO=6 THEN NO=5:RETURN
3620 IF TH(0,2)=0 THEN 4710
3630 IF CO=7 THEN PRINT "トオッテキタ ハズ゛ ノ ト゛ア カ゛ キエテシマッテ モト゛レマセン。":RETURN
3640 IF (CO=1008)*(H(3)=1)*(H(1)=0) THEN NO=3:RETURN
3650 IF CO=1416 THEN 3680
3660 IF CO=1612 THEN 3720
3670 GOTO 1920
3680 REM'*
3690 IF H(3)=1 THEN 1930
3700 H(3)=1
3710 GOTO 1910
3720 REM'*
3730 USR($0F14)
3740 RESTORE 3750:GOSUB 10550
3750 DATA "「ハーイ コンニチワ! ウサキ゛サン!」/"
3760 DATA "スルト ウサキ゛サン.../",*"
3770 IF H(9)=1 THEN 3830
3780 H(9)=1
3790 GOSUB 10550
3800 DATA "「ヤア イイトコロニ キテクレタ。 シ゛ツハ エントツ ニ ナニカカ゛ ツマッテ コマッテイルンタ゛。"
3810 DATA "ナントカシテ ソレ ヲ トリノソ゛イテクタ゛サイ。」ト イッテ イエ ノ ホウ ヲ ユヒ゛サシテ イマス。/",*"
3820 RETURN
3830 RESTORE 3840:GOSUB 10550
3840 DATA "「ナント゛モ ハナシカケナイテ゛クレ。ワタシ ハ イソカ゛シイノタ゛。」/",*"
3850 RETURN
```

```
3860 REM'*** SCENE 3
3870 H(3)=2
3880 IF CO=640 THEN 4270
3890 IF CO=938 THEN 3940
3900 IF CO=1139 THEN 4100
3910 IF CO=1829 THEN 4210
3920 IF CO=2103 THEN 4310
3930 GOTO 1920
3940 REM'*
3950 GOSUB 1060
3960 IF MID$(A$,27,1)="0" THEN 1920
3970 IF H(1)=1 THEN 4030
3980 RESTORE 3990:GOSUB 10550
3990 DATA "ｺｺﾔｯ ｱﾉ ｼｮｳｼﾞｮ ｶﾞ ｲﾅｸﾅｯﾃ ｼﾏｯﾀｿﾞ。/ｻﾃﾊ ｱﾉ ｼﾞｭｰｽ ﾗ ﾉﾝﾀﾞﾅ...ｺ/"
4000 DATA "ﾄ ｲｳﾜｹﾃ ｱﾉ ｼﾞｭｰｽ ﾊ ﾁｲｻｸﾅﾙｸｽﾘ ﾃﾞｼﾀ。/"
4010 DATA "ｿｼﾃ ｱﾅﾀ ﾊ ﾁｲｻｸﾅﾘｽｷﾞﾃ ｷｴﾃ ﾅｸﾅｯﾃ ｼﾏｯﾀﾉﾃﾞｼﾀ。/ﾒﾃﾞﾀｼ ﾒﾃﾞﾀｼ。","*"
4020 GOTO 10660
4030 GOSUB 1910
4040 H(1)=0
4050 RESTORE 4060:GOSUB 10550
4060 DATA "ｱﾅﾀ ﾊ ﾓﾄ ﾉ ｶﾗﾀﾞ ﾉ ｵｵｷｻ ﾆ ﾓﾄﾞﾙｺﾄ ｶﾞ ﾃﾞｷﾏｼﾀ。/"
4070 DATA "ｿﾉ ｼﾞｭｰｽ ﾉ ﾊｲｯﾀ ﾋﾞﾝ ﾊ ｵﾄﾓﾅｸ ｷｴﾃ ｼﾏｲﾏｼﾀ。/","*"
4080 H(0)=H(0)-1
4090 RETURN
4100 REM'*
4110 GOSUB 1060
4120 IF MID$(A$,26,1)="0" THEN 1920
4130 IF H(13)=0 THEN PRINT "ｽｽﾞｼｲ...":RETURN
4140 GOSUB 1910
4150 RESTORE 4160:GOSUB 10550
4160 DATA "ﾐﾙﾐﾙ ｳﾁﾆ ｶﾗﾀﾞ ｶﾞ ｵｵｷｸ ﾅｯﾃ ﾍﾔ ｲｯﾊﾟｲ ﾆ ﾅｯﾃ "
4170 DATA "ﾃﾌﾞｸﾛ ﾓ ｳﾁﾜ ﾓ ﾄﾞｺｶ ﾍ ｷｴﾃ ｼﾏｲﾏｼﾀ。/","*"
4180 H(1)=1
4190 H(0)=H(0)-6
4200 RETURN
4210 REM'*
4220 GOSUB 1060
4230 IF MID$(A$,27-BT,1)="0" THEN 1920
4240 IF H(13)=1 THEN PRINT "ﾓｳ ﾊﾏｯﾃｲﾏｽ。":RETURN
4250 H(13)=1
4260 GOTO 1910
4270 REM'*
4280 IF H(1)=1 THEN 1920
4290 PRINT "ﾅｾﾞｶ ｿﾄ ﾍ ﾃﾞﾗﾚﾏｾﾝ。"
4300 RETURN
4310 REM'*
4320 IF H(1)=1 THEN 1920
4330 GOSUB 1060
4340 IF MID$(A$,20,1)="0" THEN 4380
4350 PRINT "ｲﾂﾉﾏﾆｶ ｲｴ ｶﾞ ｿﾗ ﾆ ｳｶﾝﾃﾞ ｲﾏｽ。"
4360 NO=4
4370 RETURN
4380 A=TH(1,3)
4390 GOSUB 1070
4400 B$="ﾅﾆｶ"
4410 IF MID$(A$,20,1)="1" THEN B$=NA$(1,7)
4420 PRINT "ﾅｶ ﾆ ";B$;" ｶﾞ ﾂﾏｯﾃｲﾃ ﾉﾎﾞﾚﾏｾﾝ。"
4430 RETURN
4440 REM'*** SCENE 4
4450 IF (CO=2235)+(CO=1435)*(LU$="E") THEN 4490
4460 IF CO=1003 THEN NO=3:RETURN
4470 IF CO=23 THEN 4550
4480 GOTO 1920
4490 REM'*
4500 GOSUB 1060
4510 IF MID$(A$,27-BT,1)="0" THEN 1920
4520 IF H(14)=1 THEN 1930
4530 H(14)=1
4540 GOTO 1910
4550 REM'*
4560 GOSUB 1060
4570 IF (H(14)=1)*(MID$(A$,19,1)="1") THEN 4610
4580 RESTORE 4590:GOSUB 10550
4590 DATA "ｼﾞﾒﾝ ﾆ ｵﾁﾃ ｹｶﾞ ﾗ ｼﾃ ｳｺﾞｹﾅｸ ﾅｯﾃｼﾏｲﾏｼﾀ。/","*"
4600 GOTO 10660
4610 NO=2
4620 NO$(2)="ｳｻｷﾞ ﾉ ｲｴ ﾉ ｱｯﾀﾄｺﾛ"
4630 RESTORE 4640:GOSUB 10550
4640 DATA "ｳﾏｸ ｵﾘﾙｺﾄｶﾞ ﾃﾞｷﾏｼﾀ。/","/"
4650 DATA "ｳｻｷﾞ ﾉ ｲｴ ﾊ ﾄﾝﾃﾞｲｯﾃ ｼﾏｲﾏｼﾀ。/"
4660 DATA "ｳｻｷﾞ ﾓ ｲｴ ﾗ ｵｲｶｹﾃ ﾄﾞｺｶ ﾍ ｲｯﾃｼﾏｲﾏｼﾀ。/","/"
4670 DATA "ｶｻ ﾊ ﾄｼﾞﾃ ｵｷﾏｼｮｳ。/","*"
4680 H(14)=0
4690 TH(0,2)=0
4700 RETURN
4710 REM'*
4720 IF CO=7 THEN PRINT "ﾓﾄﾞﾚﾏｾﾝ。":RETURN
4730 IF CO=6 THEN NO=5:GOTO 1910
4740 REM'*** SCENE 5
4750 IF CO=6 THEN 4800
4760 IF CO=7 THEN NO=2:RETURN
4770 IF CO=334 THEN 4840
4780 IF CO=2125 THEN NO=6:GOTO 1860
4790 GOTO 1920
4800 REM'*
```

```
4810 IF H(4)=1 THEN NO=7:RETURN
4820 PRINT "オオモナ イヌ ガ シッポ ヲ フッテ イクテニ タッテイルノデ コワクテ トオレナイ。"
4830 RETURN
4840 REM'*
4850 A=TH(1,5)
4860 GOSUB 1070
4870 IF (MID$(A$,20,1)="0")+(H(4)=1) THEN RETURN
4880 PRINT "オオモナ イヌ ハ ボウモレ ヲ クワエテ トコカヘ イッチャイマシタ。"
4890 TH(0,5)=TH(0,5)-32
4900 TH(1,5)=TH(1,5)-128
4910 H(4)=1
4920 RETURN
4930 REM'*** SCENE 6
4940 IF CO=25 THEN PRINT "ウマク オリラレナイ。":RETURN
4950 IF (CO=2235)+(CO=1435)*(LU$="E") THEN 5060
4960 IF CO=23 THEN 4980
4970 GOTO 1920
4980 REM'*
4990 GOSUB 1060
5000 IF (H(14)=0)+(MID$(A$,19,1)="0") THEN 4580
5010 PRINT "チャクチ セイコウ。"
5020 NO=5
5030 PRINT "カサ ハ トジテ オモマス。"
5040 H(14)=0
5050 RETURN
5060 REM'*
5070 GOSUB 1060
5080 IF MID$(A$,27-BT,1)="0" THEN 1920
5090 IF H(14)=1 THEN 1930
5100 H(14)=1
5110 GOTO 1910
5120 REM'*** SCENE 7
5130 IF CO=6 THEN NO=8:RETURN
5140 IF CO=7 THEN NO=5:RETURN
5150 IF CO=1602 THEN 5170
5160 GOTO 1920
5170 REM'*
5180 USR($0F14)
5190 RESTORE 5200:GOSUB 10550
5200 DATA "「ハロー! アオムシサン!」/","*"
5210 IF H(10)=1 THEN H(10)=2:GOTO 5300
5220 IF H(10)=2 THEN 1920
5230 GOSUB 10550
5240 DATA "スルト マユ ヲ ツクリナガラ/"
5250 DATA "「イイコト ヲ オシエテ アゲヨウ」ト アオムシサン。/"
5260 DATA "「コノ キノコ ヲ タベテ ピーヒョロルマー ト トナエレバ オオキクナリ "
5270 DATA "「パーパーレホー ト トナエレバ チイサク ナレル。」ト オシエテ クレマシタ。/","*"
5280 H(10)=1
5290 RETURN
5300 REM'*
5310 RESTORE 5320:GOSUB 10550
5320 DATA "アオムシ サン ハ モスラ ニ ナッテ トンデ イッテ シマイマシタ。/","*"
5330 TH(0,7)=0
5340 RETURN
5350 REM'*** SCENE 8
5360 IF CO=6 THEN NO=9:H(15)=1:RETURN
5370 IF CO=7 THEN NO=7:RETURN
5380 IF (CO=1019)*(H(5)=1)*(H(1)=0) THEN NO=11:RETURN
5390 IF CO=1406 THEN 5420
5400 IF CO=1600 THEN PRINT "ヒトリ デ ブツブツ イッテマス。":RETURN
5410 GOTO 1920
5420 REM'*
5430 IF H(5)=1 THEN 1930
5440 H(5)=1
5450 GOTO 1910
5460 REM'*** SCENE 9
5470 IF CO=6 THEN 5530
5480 IF CO=7 THEN 5820
5490 IF CO=2123 THEN 5670
5500 IF CO=2441 THEN 5720
5510 IF CO=1441 THEN 5770
5520 GOTO 1920
5530 REM'*
5540 IF H(15)=2 THEN NO=12:RETURN
5550 IF H(15)=1 THEN PRINT "カヘ ニ サエモ ラレテ ススメマセン。":RETURN
5560 RESTORE 5570:GOSUB 10550
5570 DATA "「アレッ? トウリヌケチャッター。」/"
5580 DATA "トウヤラ アレハ アンジニヨッテ ツクリダサレタ カクウノ カヘ ダッタヨウデス。/"
5590 DATA "「サスカ ワンダーラント ダワ!"
5600 DATA "コノハナシラモイタラ SFファンノ オネエサンカ ヨロコブデショウネー キット。/","/"
5610 DATA "ナト ト カンカエテイルウチニ ソノカヘハ ナクナッテシマイマシタ。/"
5620 DATA "トウヤラ アンジカ トケタヨウデス。/","*"
5630 TH(0,9)=TH(0,9)-TA(24)
5640 H(15)=2
5650 NO=12
5660 RETURN
5670 REM'*
5680 IF H(15)=0 THEN PRINT "「メラ ツブッテナンテ ノボレマセンヨ!」":RETURN
5690 PRINT "ナントカ 3バンメ ノ エダマデ ノボルコトカ デキマシタ。"
5700 NO=10
5710 RETURN
5720 REM'*
5730 IF H(15)=2 THEN 1920
5740 IF H(15)=0 THEN PRINT "ステニ トジテ イマス。":RETURN
5750 H(15)=0
```

```
5760 GOTO 1910
5770 REM'*
5780 IF H(15)=2 THEN 1920
5790 IF H(15)=1 THEN 1930
5800 H(15)=1
5810 GOTO 1910
5820 REM'*
5830 IF H(15)=1 THEN NO=8:GOTO 1910
5840 PRINT "マワリ ノ ヨウス カﾞ ヨク ミエナイノﾃﾞ ウゴﾞケマセン。"
5850 RETURN
5860 REM'*** SCENE 10
5870 IF H(1)=1 THEN 1920
5880 IF CO=23 THEN 5920
5890 IF (CO=2235)+(CO=1435)*(LU$="E") THEN 6010
5900 IF CO=25 THEN PRINT "シタ ヲ ミタラ コウカアッテ オリレナイ。":RETURN
5910 GOTO 1920
5920 REM'*
5930 GOSUB 1060
5940 IF (H(14)=0)+(MID$(A$,19,1)="O") THEN 4580
5950 RESTORE 5790:GOSUB 10550
5960 DATA "マタ サッキ ノ トコロ ニ オリテ キマシタ。/"
5970 DATA "カサ ヲ トシﾞテ オキマス。/","*"
5980 NO=9
5990 H(14)=0
6000 RETURN
6010 REM'*
6020 IF H(14)=1 THEN 1930
6030 H(14)=1
6040 GOTO 1910
6050 REM'*** SCENE 11
6060 IF (CO=640)*(H(1)=0) THEN NO=8:RETURN
6070 IF CO=1610 THEN 6090
6080 GOTO 1920
6090 REM'*
6100 USR($0F14)
6110 RESTORE 6120:GOSUB 10550
6120 DATA "「アノー...」ト ヒカエメ ナ アリスワン テﾞスカﾞ .../","*"
6130 IF H(11)=1 THEN 6230
6140 GOSUB 10550
6150 DATA "「アー ウルサイ ウルサイ。アンタ コレモッテ トット ト オユキ!」"
6160 DATA "ト イッテ コウシャクフジﾞン ハ アカンボﾞウ ヲ アリス ニ ムカッテ ホウリナゲﾞマシタ。/"
6170 DATA "シハﾞラク ソコテﾞ ウロウロ シテイタラ ナント アカンボﾞウ カﾞ ミルミル コブﾞタ ヘト "
6180 DATA "ソシテ ツイニ ハ オノ ニ カワッテ シマイマシタ。/"
6190 DATA "フシギﾅ コトッテ アルモンテﾞスネェ.../","*"
6200 H(0)=H(0)+16
6210 H(11)=1
6220 RETURN
6230 RESTORE 6240:GOSUB 10550
6240 DATA "ヒﾟーヒﾟー キャーキャー テﾞ ナニ ヲ イッテイルノカ サッパﾘ ワカリマセン。/","*"
6250 RETURN
6260 REM'*** SCENE 12
6270 IF CO=6 THEN NO=13:RETURN
6280 IF CO=7 THEN NO=9:RETURN
6290 IF (CO=1537)+(CO=1601) THEN 6330
6300 IF CO=1231 THEN 6350
6310 GOTO 1920
6320 REM'*
6330 PRINT "「ニャハハハハハ!」 ヒョーキン ﾅ ネコテﾞス。"
6340 RETURN
6350 REM'*
6360 GOSUB 1060
6370 IF MID$(A$,27-BT,1)="O" THEN PRINT "モッテ イマセン。":RETURN
6380 GOSUB 1910
6390 RESTORE 6410:GOSUB 10550
6400 USR($0F14)
6410 DATA "「コレ アゲﾞマスｳ。」ト アリスチャン。/"
6420 DATA "「トﾞーモ アリガﾞト。オレイニ イイコト オシエテアルヨ。」 ト ネコサン。/"
6430 DATA "「サンカﾞツ ウサギﾞ ノ トコロ ニ イクト オモシロイヨ。コノサキ ノ モリヲ "
6440 DATA "キタ,ヒガﾞシ,キタ,ヒガﾞシ,ミナミ ノ ジﾞュン ニ ススム ト ソイツ ノ イエ ノ マエニ テﾞル。"
6450 DATA "クレグﾞレモ モリ ノ ナカテﾞ マヨウナヨ ニトﾞト ヌクラレナク ナルゾﾞ。/","*"
6460 H(0)=H(0)-16
6470 RETURN
6480 REM'*** SCENE 13
6490 RESTORE 6500:GOSUB 10550
6500 DATA "ミチカﾞ メイロ ノ ヨウニ イリクンテﾞ イルノテﾞ タイヨウ ヲ タヨリ ニ "
6510 DATA "N(キタ),S(ミナミ),W(ニシ),E(ヒガﾞシ) テﾞ ススミマス。/","*"
6520 D$="NENES":C$=""
6530 INPUT "トﾞッチ ノ ホウコウ";A$
6540 GOSUB 1140 ―――――――――――――――――――――――――― 6540　必要ありません
6550 IF A$="N")+(A$="S")+(A$="W")+(A$="E" THEN 6580
6560 PRINT "???"
6570 GOTO 6530
6580 C$=C$+A$
6590 IF C$=D$ THEN 6650
6600 IF LEN(C$)<59 THEN 6530
6610 GOSUB 10550
6620 DATA "トツゼﾞン テﾞスカﾞ アリス カﾞ カロウ ノ タメ タオレテ シマイマシタ。/"
6630 DATA "シタカﾞッテ コノ ケﾞーム モ ジﾞエントﾞ ニ ナルノテﾞシタ。/","*"
6640 GOTO 10660
6650 FOR I=0 TO 6001:NEXT
6660 RESTORE 6670:GOSUB 10550
6670 DATA "トツゼﾞン カラダﾞ カﾞ ヒカリ ニ ツツマレ "
6680 DATA "オチハﾞ カﾞ マイアカﾞリ マワリ ノ クシキカﾞ ナカﾞレテイキマス。/","*"
6690 FOR I=0 TO 6001:NEXT
6700 NO=14
```

```
6710 RETURN
6720 REM'*** SCENE 14
6730 IF CO=6 THEN NO=16:RETURN
6740 IF CO=7 THEN PRINT "モウ モト゛レマセン。":RETURN
6750 IF (CO=1020)*(H(6)=1)*(H(1)=0) THEN NO=15:RETURN
6760 IF CO=1417 THEN 6780
6770 GOTO 1920
6780 REM'*
6790 IF H(6)=1 THEN 1930
6800 H(6)=1
6810 GOTO 1910
6820 REM'*** SCENE 15
6830 IF (CO=640)*(H(1)=0) THEN NO=14:RETURN
6840 IF CO=836 THEN 6870
6850 IF (CO=1537)+(CO=1621) THEN 6950
6860 GOTO 1920
6870 REM'*
6880 IF H(16)=0 THEN 1920
6890 IF H(17)=0 THEN 6920
6900 PRINT "ダ゛レカ ト ネッシン ニ シャベ゛ッテ イテ ハナシ ヲ キイテクレマセン。"
6910 RETURN
6920 PRINT "「ソレハ ´ツキカ゛ トッテモ アオイカラ´ ナノダ゛。」"
6930 H(17)=1
6940 RETURN
6950 REM'*
6960 USR($0F14)
6970 RESTORE 6980:GOSUB 10550
6980 DATA "「コンニチハ! ブ゛ンカ シマセンカ ?」/","*"
6990 IF H(16)=1 THEN 6900
7000 GOSUB 10550
7010 DATA "「アハハハハ! ヒトツ ワタシ カ゛ オモシロイ クイズ゛ ヲ ダ゛ソウ。」ト ウサギ゛サン。/"
7020 DATA "「ナゼ゛ ツクエ ハ ヒルマ ニ クシャミ ヲ スルカ? ト゛ウテ゛ス ワカラナイテ゛ショウ。アハハ!」/"
7030 DATA "ウーン! ワケノワカラナイ ヒト... チカ゛ッタ ウサギ゛サンダ゛コト。"
7040 DATA "アリス アタマ カ゛ イタク ナッチャウ。/","*"
7050 H(16)=1
7060 RETURN
7070 REM'*** SCENE 16
7080 IF CO=6 THEN PRINT "ココ カ゛ イキト゛マリ テ゛ス ":RETURN
7090 IF CO=7 THEN NO=14:RETURN
7100 IF (CO=1014)*(H(7)=1) THEN NO=17:RETURN
7110 IF CO=1418 THEN 7130
7120 GOTO 1920
7130 REM'*
7140 IF H(7)=1 THEN 1930
7150 IF H(1)=0 THEN PRINT "ト゛ア カ゛ オオキスギ゛テ アケルコトカ゛ テ゛キマセン。":RETURN
7160 H(7)=1
7170 GOTO 1910
7180 REM'*** SCENE 17
7190 IF CO=230 THEN 7300
7200 IF CO=640 THEN NO=16:RETURN
7210 IF (CO=1022)*(H(8)=3)*(H(1)=0) THEN NO=18:RETURN
7220 IF CO=1330 THEN 7260
7230 IF CO=1422 THEN 7400
7240 IF CO=1722 THEN 7440
7250 GOTO 1920
7260 REM'*
7270 A=TH(1,17):GOSUB 1070
7280 IF MID$(A$,27-BT,1)="0" THEN PRINT "ソンナモノ ハ ニャーダ゛ギ゛ャー。":RETURN
7290 GOSUB 1910
7300 IF H(8)=1 THEN RETURN
7310 RESTORE 7320:GOSUB 10550
7320 DATA "アッ! コンナトコロニ ","*"
7330 PRINT NA$(0,22);
7340 GOSUB 10550
7350 DATA " カ゛ アルゥー!!/"
7360 DATA " シカシ ナンテ゛コンナトコロニ ト゛ア カ゛ アルンテ゛ショーネー! アリス アタマカ゛ イタァーイノ!/","*"
7370 H(8)=1
7380 TH(0,17)=TH(0,17)+4194304
7390 RETURN
7400 REM'*
7410 ON H(8)+1 GOTO 1920,1950,7420,1930
7420 H(8)=3
7430 GOTO 1910
7440 REM'*
7450 ON H(8)+1 GOTO 1920,7460,1940,1930
7460 GOSUB 1060
7470 IF MID$(A$,22,1)="0" THEN 1960
7480 H(8)=2
7490 GOTO 1910
7500 REM'*** SCENE 18
7510 IF CO=1626 THEN 7560
7520 IF (CO=2626)*(H(18)=1) THEN 7650
7530 IF CO=6 THEN PRINT "ト゛コ ヘ イクノカ ワカラナイ。":RETURN
7540 IF CO=7 THEN PRINT "ト゛コカラモタノカ ワスレテ シマイマシタ。":RETURN
7550 GOTO 1920
7560 REM'*
7570 IF H(18)=1 THEN PRINT "スミ ヲ ハキカケラレマシタ。":RETURN
7580 RESTORE 7590:GOSUB 10550
7590 DATA "タ コ:「モウカリマッカ ?」/"
7600 DATA "アリス:「マ。マア ボ゛チボ゛チ テ゛スゥ。 アンサントコハ ト゛ナイテ゛スカ ?」/"
7610 DATA "タ コ:「ソフトノ ウリアケ゛カ゛ イマイチテ゛... グカラヒ゛ル アト゛ヘ゛ンチャー モ ヨロシク。/"
7620 DATA " ナト゛ト イイナカ゛ラ タコ ハ アルキタ゛シマシタ。/","*"
7630 H(18)=1
7640 RETURN
7650 REM'*
```

```
7660 RESTORE 7670:GOSUB 10550
7670 DATA " タコ ハ トﾞントﾞン アルイテユキマス。/"
7680 DATA "オワレナイヨウニ イソギﾞアシ テﾞ ツイテイキマス。/"
7690 DATA "キカﾞツクト イツノマニカ モリ ノ ソト ニ テﾞテ イマシタ。/"
7700 DATA "アタリ イチメン オハナハﾞタケ テﾞ タコ ハ トﾞコカニ イッテシマイマシタ。/","*"
7710 PRINT
7720 NO=19
7730 RETURN
7740 REM****** SCENE 19
7750 IF CO=6 THEN GOSUB 1910:PRINT "アレッ! サッキト オンナジﾞ ハﾞショダﾞ!!":RETURN
7760 IF (CO=1537)+(CO=1609)+(CO=1611) THEN 7780
7770 GOTO 1920
7780 REM*
7790 IF H(1)=0 THEN 8020
7800 RESTORE 7810:GOSUB 10550
7810 DATA "「コ.コンニチワ...アハハッ」 アリスチャン ナセﾞカ アカﾞッテ イマス。/","*"
7820 IF H(12)=1 THEN GOSUB 10550:RETURN
7830 DATA "モウ アイテニ シテクレマセンデﾞス。 カナシイ..../","*"
7840 H(12)=1
7850 RESTORE 7860:GOSUB 10550
7860 DATA "「ウーン! ミカケナイ カオダﾞナー」ト オオサマ。「キミ ハ ナンテイウ ナマエダﾞ?」/"
7870 DATA "ナンテ キカレテ イマス。ココハ コタエテ オイタホウカﾞ ヨサソウ テﾞスヨ?/","*"
7880 INPUT "ナマエ ハ ";A$
7890 GOSUB 1140 ―――――――――――――――――――――――――――――――7890　必要ありません
7900 FOR I=0 TO 2000:NEXT
7910 IF (LU$="J")*(A$="アリス")+(A$="ALICE") THEN 7960
7920 GOSUB 10550
7930 DATA "「アッ ソウイエバﾊﾞ ソウダﾞッタナ。 チカゴロ キオクリョク カﾞ オチテナー! アッハッハ!!」"
7940 DATA "ナ ナント! ダﾞレカト マチカﾞエラレテ シマイマシタ。/","*"
7950 RETURN
7960 RESTORE 7970:GOSUB 10550
7970 DATA "「ウーン! ソンナ マトモナ ナマエ ノ モノ ハ シラナイナー。 チョット イッショニ キタマエ。」"
7980 DATA "ト イッテ ヒッパﾟラレテ イキマシタ。 コノ オオサマ ロリコンテﾞ ナイテﾞショーネー..../","*"
7990 NO=20
8000 RETURN
8010 REM*
8020 PRINT "カラダﾞ カﾞ チイサイノテﾞ セーンセーン キカﾞツカナイ ヨウデﾞス。"
8030 RETURN
8040 REM****** SCENE 20
8050 IF CO<>1609 THEN 1920
8060 RESTORE 8070:GOSUB 10550
8070 DATA "「....ホホウ ソウスルト オマエハ コノクニノ ジﾞュウミンデﾞハ ナイノダﾞナ。」/"
8080 DATA "「エエ ソウナノ。ウサギﾞサン ヲ オイカケテキタラ イツノマニカ ココヘ モチシマッタンデﾞス。トﾞウヤッタラ"
8090 DATA " オウチヘ カエレルノカ オシエテクレナイカシラ。"
8100 DATA "オチャノ ジﾞカンニ オクレルト オカアサマニ シカラレチャウ。」/"
8110 DATA "「ウーン ワシニハ ヨクワカランカﾞ ワシノトコロニハ"
8120 DATA " モノヲ ヨクシッテイル モノカﾞ イルカラ ソイツニ キイテミヨウ。"
8130 DATA "オイ OTOMIHSAH ヲ ヨンデﾞコイ。」/"
8140 DATA "「オヨビﾋﾞデﾞショウカ。」/"
8150 DATA "「コノコ カﾞ ウチニカエルニハ トﾞウシタラ ヨイノカ オシエテ ヤッテクレ。」/"
8160 DATA "「ソレデﾞハ ﾞナセﾞ ツクエ ハ ヒルマ ニ クシャミ ヲ スルカﾞ。"
8170 DATA "コノ ナソﾞナソﾞニ コタエラレタラ オマエノ スンデﾞイタ セカイヘ カエルミチヲ オシエテヤロウ。」/","*"
8180 INPUT "コタエハ ";A$
8190 B$=""
8200 FOR I=1 TO LEN(A$)
8210 IF MID$(A$,I,1)<>" " THEN B$=B$+MID$(A$,I,1)
8220 NEXT
8230 IF B$="ツキカﾞトッテモアオイカラ" THEN 8320
8240 RESTORE 8250:GOSUB 10550
8250 DATA "「サﾞンネンデﾞシタ。モウ シラナイ。」/"
8260 DATA "ト,イウワケテﾞ アリス チャン ハ ウチヘ カエレナクナッテシマイ ソレラミタ オオサマハ"
8270 DATA "カワイソウニ オモッテ アリス ヲ オシロ ノ ハﾟンヤキ ニ シテクレマシタ。/"
8280 DATA " モトモト アリスハ ハﾟンラヤクノカﾞ スキダﾞッタノテﾞ ハﾟンラヤキナカﾞラ シアワセニ クラシマシタ。/"
8290 DATA "コノハ ノ ハナシコﾞエ カﾞ キコエルヨウナ サミシイ ヨル ニ アリス ハ オカアサマ ヤ オネエサマ ノ"
8300 DATA "コト ヲ オモイダﾞシテ ソット ナミダﾞ ヲ ナガﾞスノテﾞシタ。/","*"
8310 GOTO 10660
8320 RESTORE 8330:GOSUB 10550
8330 DATA "「オオアタリ! ソレデﾞハ キミノ ウチノ マエマテﾞ オワッテイッテ アケﾞヨウ。コチラヘ キナサイ。」/"
8340 DATA "アリス ハ ソノ OTOMIHSAH ニ ツイテ ユキマシタ。スルト トツゼﾝ シロイ ヌノ ノ ヨウナ "
8350 DATA "モノカﾞ カブﾞサッテキマシタ。OTOMIHSAH ハ トﾞンドﾝ サキヘト イッテシマウ ノテﾞ "
8360 DATA "「マッテーー!」ト ヨンダﾞノ テﾞスカﾞ キコエナイヨウデﾞス。"
8370 DATA "ダﾞンダﾞン マワリ ノ ケシキカﾞ ウスレテユクヨウニ カンジﾞマシタ。/","*"
8380 PRINT
8390 REM****** FINAL
8400 FOR I=0 TO 8201:NEXT
8410 RESTORE 8420:GOSUB 10550
8420 DATA " アリス ハ トﾞテ ノ ウエ テﾞ オネエサン ノ ヒザﾞ ニ アタマ ヲ ノセテ ネテイタ ジﾞブﾝ ニ "
8430 DATA "キカﾞツキマシタ。 オネエサン ハ アリス ノ カミノケヲ ヤサシク テナﾞテイマシタ。/"
8440 DATA "「オキナサイ アリス!」オネエサン カﾞ ミミモトテﾞ ササヤキマス。「ホントウニ ヨク ネムッタワネェ!」/"
8450 DATA "「ア オネエサン。ワタシ トッテモ オカシナ ユメ ヲ ミタノ!"
8460 DATA "ウサギﾞサン カﾞ イテ オオサマ ヤ ジﾞョオウサマ カﾞ イテ...."","/"
8470 DATA " アリス ハ ソウイッテ オネエサン ニ オモイダﾞセルダﾞケ セイイッパﾟイ イママテﾞニ"
8480 DATA " タイケン シタ カノジﾞョ ノ フシギﾞナ ホﾞウケン ノ ハナシ ヲ シテ キカセタノテﾞス。","/"
8490 DATA " アリス カﾞ ハナシオワル ト オネエサン ハ カノジﾞョ ニ キス シテ イイマシタ。/"
8500 DATA "「ホントウニ フシギﾞナ ユメダﾞッタワネ テﾞモ モウ イソイデﾞ オチャ ニ イキナサイ。"
8510 DATA " モウ オソク....」/"
8520 DATA " ソコデﾞ オネエサン ハ コトバﾞ ヲ ウシナイマシタ。"
8530 DATA " アリス ハ ソノ オネエサン ノ ヒキツッタ カオヲ ミテ オモワズ ウシロヲ フリカエリマシタ。/"
8540 DATA "ナント ソコニハ オオキナ シロクジﾞラ カﾞ コチラヲ ムイテ テヲ フッテ イルデﾞハ"
8550 DATA "アリマセンカ!! ソシテ ユックリト クチヲ ヒライテ コウイイマシタ。/","/"
8560 DATA " 「イッショ ニ サンポ ニ テﾞカケマセンカ?」/","*"
8570 PRINT
8580 FOR I=0 TO 6001:NEXT
8590 GOSUB 10550
8600 DATA "コンドﾞ ハ アナタ カﾞ ADVENTURE ヲ ツクル バﾝ デﾞスヨ!!","*"
```

```
8610 END
8620 REM'*** INIT
8630 CONSOLE S 0,24:CONSOLE C 40:PRINT CHR$(6)――8630  CONSOLE 0, 25, 0, 40 : COLOR  ,,6,0
8650 PRINT CHR$(6)
8660 DIM NA$(1,26),VR$(26),MA$(20),CH(4),TA(26),H(21)
8670 DIM TR$(1,26),FM$(35),TH(1,20),NO$(20),A$(25)
8680 PRINT:PRINT "(1)ﾆﾎﾝｺﾞ(ｶﾀｶﾅ)  (2)ｴｲｺﾞ"
8690 PRINT:PRINT " 1 ﾏﾀﾊ 2 ﾉ ｽｳｼﾞ ﾗ ｵｼﾃｸﾀﾞｻｲ。"
8700 PRINT:PRINT "       ";
8710 GET A$:IF A$="" THEN 8710
8720 IF (A$<>"1")*(A$<>"2") THEN 8710
8730 IF A$="1" THEN LU$="J":PRINT "ﾆﾎﾝｺﾞ":GOTO 8940
8740 LU$="E":PRINT "ｴｲｺﾞ" :RESTORE 8780
8750 FOR I=1 TO 26
8760 READ VR$(I)
8770 NEXT
8780 DATA LOOK,TAKE,THROW,EAT,INVENTORY,GO,BACK,ASK,DRINK
8790 DATA ENTER,FAN,GIVE,MOVE,OPEN,SAY,TALK,UNLOCK,WEAR
8800 DATA SAVE,LOAD,CLIMB,OPEN,JUMP,CLOSE,DOWN,FOLLOW
8810 FOR I=0 TO 26
8820 READ NA$(0,I)
8830 NEXT
8840 DATA BUTLER,CAT,CATAPILLAR,CHIMNEY,CURTAIN,DOG,DOOR,DRESSER,HOUSE
8850 DATA KING,MARCHIONESS,QUEEN,RABBIT,SMALL DOOR,STORE,TABLE
8860 DATA DOOR,DOOR,DOOR,HOUSE,HOUSE,RABBIT,SMALL DOOR,TREE,WALL
8870 DATA TREE,TAKO
8880 FOR I=0 TO 14
8890 READ NA$(1,I)
8900 NEXT
8910 GOTO 9120
8920 DATA BOTTLE,FAN,GLOVE,HAT,HATCHET,KEY,MUSHROOM,STICK,PARASOL
8930 DATA ANSWER,HELLO,JUICE,MYSELF,OUT,EYE
8940 RESTORE 8980
8950 FOR I=1 TO 26
8960 READ VR$(I)
8970 NEXT
8980 DATA ﾐﾙ,ﾄﾙ,ﾅｸﾞﾙ,ﾀﾍﾞﾙ,ﾓﾁﾓﾉ,ｲｸ,ﾓﾄﾞﾙ,ｷｸ,ﾉﾑ
8990 DATA ﾊｲﾙ,ｱｵｸﾞ,ｱｹﾞﾙ,ｳｺﾞｶｽ,ｱｹﾙ,ｲｳ,ﾊﾅｽ,ｶｷﾞﾗｱｹﾙ,ﾊﾒﾙ,ｾｰﾌﾞ,ﾛｰﾄﾞ
9000 DATA ﾉﾎﾞﾙ,ｻｽ,ﾄﾋﾞｵﾘﾙ,ﾄｼﾞﾙ,ｵﾘﾙ,ﾂｲﾃｲｸ
9010 FOR I=0 TO 26
9020 READ NA$(0,I)
9030 NEXT
9040 DATA ｼﾂｼﾞ,ﾈｺ,ｲﾓﾑｼ,ｴﾝﾄﾂ,ｶｰﾃﾝ,ｲﾇ,ﾄﾞｱ,ｷｮｳﾀﾞｲ,ｲｴ
9050 DATA ｵｳｻﾏ,ｺｳｼｬｸﾌｼﾞﾝ,ｼﾞｮｵｳ,ｳｻｷﾞ,ﾁｲｻｲﾄﾞｱ,ﾐｾ,ﾃｰﾌﾞﾙ
9060 DATA ﾄﾞｱ,ﾄﾞｱ,ﾄﾞｱ,ｲｴ,ｲｴ,ｳｻｷﾞ,ﾁｲｻｲﾄﾞｱ,ｷ,ｶﾍﾞ,ｷ,ﾀｺ
9070 FOR I=0 TO 14
9080 READ NA$(1,I)
9090 NEXT
9100 DATA ﾋﾞﾝ,ﾂﾁﾜ,ﾃﾌﾞｸﾛ,ﾎﾞｳｼ,ｵﾉ,ｶｷﾞ,ｷﾉｺ,ﾎﾞｳ,ﾊﾟﾗｿﾙ
9110 DATA ｺｹﾞｴ,ｺﾝﾆﾁﾜ,ｼﾞｭｰｽ,ｼﾞﾌﾞﾝ,ｵﾓﾃ,ﾒ
9120 RESTORE 9160
9130 FOR I=0 TO 35
9140 READ FM$(I)
9150 NEXT
9160 DATA ﾊﾝﾌﾞﾝ ｻｶﾅ ﾉ ｶｯｺｳ ﾗ ｼﾀﾞ,ｷ ﾆ ﾉﾎﾞｯﾃ ｽﾜｯﾃｲﾙ ﾆﾔｹｶﾞｵ ﾉ
9170 DATA ﾏﾙﾃﾞ ﾊｼﾓﾄ ｻﾝ ﾉ ﾖｳﾅ,ｱｶｲ ﾚﾝｶﾞﾂﾞｸﾘ ﾉ,ﾁｲｻ ﾅ ｽｶｲﾌﾞﾙｰ ﾉ
9180 DATA ﾏｯｼﾛﾅ ﾏﾙｯｺｲ,ｷﾃﾞ ﾃﾞｷﾀ ｶﾞﾝｼﾞｮｳ ｿｳﾅ,ｽｺｼ ｵｵｷｲ ｸﾗｲ ﾉ
9190 DATA ｼｬﾚﾀ ﾁｲｻﾅ,ﾀｸｻﾝ ﾉ ｸﾗｲ ﾗ ﾋｷﾂﾚﾀ
9200 DATA ﾋｽﾃﾘｯｸ ﾃﾞ ﾃﾞﾋﾞﾙｸｲｰﾝ ﾉ ﾖｳﾅ,ｾ ｶﾞ ﾀｶｸﾃ ｳﾂｸｼｲ
9210 DATA ﾄﾝﾎﾞﾒｶﾞﾈ ﾗ ｶｹ ｶｲﾁｭｳﾄﾞｹｲ ﾗ ﾓｯﾀ,5ｲﾝﾁ ﾃﾞｲｽｹｯﾄ ｸﾞﾗｲ ﾉ
9220 DATA ﾄﾞｳﾔﾗ ﾎﾞｳｼﾔ ﾉ ﾖｳﾅ,ｳﾂｸｼｸ ｽｷﾄｵｯﾀ ｶﾞﾗｽ ﾉ,ｼｬﾚﾀ ﾂｸﾘ ﾉ
9230 DATA ｺﾌｳ ﾅ,ﾊﾝｴﾝｸｲ ﾗ ｼﾀ,ﾘｯﾊﾟﾅ ﾂｸﾘﾗ ｼﾀ
9240 DATA ｳｻｷﾞ ﾉ ﾐﾐ ﾉ ﾖｳﾅ ｴﾝﾄﾂ ﾉ ﾀｯﾃｲﾙ,3ｶﾞﾂ ｳｻｷﾞ ﾄ ﾖﾊﾞﾚﾃｲﾙ
9250 DATA ﾐﾀｺﾄ ﾉ ｱﾙﾖｳﾅ,ﾀｶｲ ﾓﾐ ﾉ,ﾄﾞｺﾏﾃﾞﾓ ﾂﾂﾞｸ ｷﾞﾝｲﾛ ﾉ,ﾌﾄｲ ｴｻﾞ ﾉ,ﾀｺ ﾉ
9260 DATA ﾗﾍﾞﾙ ﾆ 「Drink me!」ﾄ ｶｲﾃｱﾙ ｼﾞｭｰｽ ﾉ ﾊｲｯﾀ
9270 DATA HUDSON ｳﾁﾜ(ｼﾙﾋﾄｿﾞ ｼﾙ)ﾆ ﾆﾀ,ｶﾀｶﾎｳ ﾀﾞｹ ﾉ ｷｯﾄﾞ ﾉ
9280 DATA ｳｻｷﾞﾖｳ ﾆ ｱﾅ ﾉ ｱｲﾀ,ﾓﾄ ﾊ ｱｶﾝﾎﾞｳ ﾉ ｽｶﾞﾀ ﾗ ｼﾃｲﾀ ﾌｼｷﾞﾅ,ｷﾝｲﾛ ﾉ
9290 DATA ﾏﾂﾀｹ ﾆ ｿｯｸﾘﾅ,ﾐﾝｷｰｽﾃｯｷ ﾆ ｿｯｸﾘﾅ,ｼﾛｲ
9300 FOR I=0 TO 26
9310 READ TR$(0,I)
9320 NEXT
9330 DATA 00*,01*,02*,03107*,04*,05*,06*,07100102*,08*,09*,10*,11*
9340 DATA 12*,13*,14*,15100105*,16*,17*,18105*,19*,20*,21*,22*
9350 DATA 23*,24*,25*,26*
9360 FOR I=0 TO 9
9370 READ TR$(1,I)
9380 NEXT
9390 DATA 27*,28*,29*,30*,31*,32*,33*,34*,35*,36*
9400 FOR I=1 TO 20
9410 READ TH(0,I),TH(1,I)
9420 NEXT
9430 DATA 32848,0,69888,0,136,2,8,0,33554464,0,0,256,4,64
9440 DATA 524353,0,25165824,0,0,0,1024,0,2,0,0,0,1179648,0
9450 DATA 2097152,0,278528,0,0,8,67108864,0,2560,0,2560,0
9460 FOR I=1 TO 20
9470 READ NO$(I)
9480 NEXT
9490 DATA ﾋﾛｲﾎｰﾙ,ｳｻｷﾞ ﾉ ｲｴ ﾉ ﾏｴ,ｳｻｷﾞ ﾉ ｲｴ ﾉ ﾅｶ,ｳｻｷﾞ ﾉ ｲｴ ﾉ ﾔﾈ ﾉ ｳｴ
9500 DATA ﾓﾘ ﾉ ﾅｶ,ｷ ﾉ ｳｴ,ﾓﾘ ﾉ ｽｺｼ ﾋﾗｹﾀ ﾄｺﾛ
9510 DATA ﾓﾘ ﾗ ﾃﾞﾀﾄｺﾛ ﾆ ｱﾙ ｺｳｼｬｸﾌｼﾞﾝ ﾉ ｲｴ ﾉ ﾏｴ,ﾐｽﾞｳﾐ ﾉ ﾎﾄﾘ
9520 DATA ﾐｽﾞｳﾐ ﾉ ﾎﾄﾘ ﾆ ｱﾙ ｷ ﾉ ｳｴ,ﾅﾍﾞ ﾔ ﾌﾗｲﾊﾟﾝ ﾉ ﾄﾋﾞｶｳ ｲｴ ﾉ ﾅｶ
9530 DATA ｼｬｰｳｯﾄ ﾉ ﾓﾘ ﾉ ｲﾘｸﾞﾁ,ｳｯｿｳ ﾄ ｼﾀ ﾓﾘ ﾉ ﾅｶ
9540 DATA ｻﾝｶﾞﾂｳｻｷﾞ ﾉ ｲｴ ﾉ ﾏｴ,ｵﾁｬｶｲ ﾉ ｻｲﾁｭｳ ﾉ ｻﾝｶﾞﾂｳｻｷﾞ ﾉ ｲｴ ﾉ ﾅｶ
9550 DATA ｵｵｷﾅ ｷﾘｶﾌﾞ ﾉ ｶﾀﾁ ﾗ ｼﾀ ﾎﾞｳｼﾔ ﾉ ﾏｴ,ﾎﾞｳｼﾔ ﾉ ﾃﾝﾅｲ
9560 DATA ﾀｺﾂﾎﾞ ﾉ ﾅｶ,ｷﾚｲﾅ ｵﾊﾅ ｶﾞ ﾀｸｻﾝ ｻｲﾃｲﾙ ｵﾊﾅﾊﾞﾀｹ
```

```
9570 DATA ﾘｯﾊﾟ ﾅ ｷｭｳﾃﾞﾝ ﾉ ﾅｶ
9580 FOR I=1 TO 20
9590 READ MA$(I)
9600 NEXT
9610 DATA 0,06,0,0,23,0,0,08,0,0,0,0,0,060B1619,12,061617,13,0,0,0
9620 FOR I=0 TO 26
9630 READ TA(I)
9640 NEXT
9650 DATA 1,2,4,8,16,32,64,128,256,512,1024,2048,4096,8192,16384
9660 DATA 32768,65536,131072,262144,524288,1048576,2097152,4194304
9670 DATA 8388608,16777216,33554432,67108864
9680 NO=1
9690 CH(4)=99
9700 H(1)=1
9710 H(15)=1
9720 H(19)=5
9730 PRINT CHR$(6)
9740 RETURN
9750 REM'*** SAVE
9760 PRINT:PRINT "ｲﾏｶﾗ ｹﾞｰﾑ ﾉ ﾄﾁｭｳｹｲｶ ｦ SAVE ｼﾏｽ。"
9770 REM'
9780 PRINT:PRINT "ﾅﾆｶ ｷｰ ｦ ｵｽﾄ SAVE ｶﾞ ﾊｼﾞﾏﾘﾏｽ。"
9790 GET A$:IF A$="" THEN 9790
9800 WOPEN "ALICE DATA"
9810 PRINT:PRINT "ﾀﾀﾞｲﾏ SAVE ﾁｭｳ ﾃﾞｽ。"
9860 FOR I=1 TO 20:PRINT/T TH(0,I):NEXT
9870 FOR I=1 TO 20:PRINT/T TH(1,I):NEXT
9880 FOR I=0 TO 26:PRINT/T TR$(0,I):NEXT
9890 FOR I=0 TO 21:PRINT/T H(I):NEXT'
9900 PRINT/T,NO
9920 CLOSE
9930 REM'
9940 PRINT:PRINT "ｼｭｳﾘｮｳ ｼﾏｼﾀ。":PRINT
9950 CO=1:GOSUB 1970
9960 GOTO 1220
9970 REM'*** LOAD
9980 PRINT:PRINT "ｲﾏｶﾗ ｹﾞｰﾑ ﾉ ﾃﾞｰﾀ ｦ LOAD ｼﾏｽ。"
9990 REM'
10000 PRINT:PRINT "ﾅﾆｶ ｷｰ ｦ ｵｽﾄ LOAD ｶﾞ ﾊｼﾞﾏﾘﾏｽ。"
10010 GET A$:IF A$="" THEN 10010
10020 ROPEN "ALICE DATA"
10030 PRINT:PRINT "ﾀﾀﾞｲﾏ LOAD ﾁｭｳ ﾃﾞｽ。"
10040 FOR I=1 TO 20:INPUT/T TH(0,I):NEXT
10050 FOR I=1 TO 20:INPUT/T TH(1,I):NEXT
10060 FOR I=0 TO 26:INPUT/T TR$(0,I):NEXT
10070 FOR I=0 TO 21:INPUT/T H(I):NEXT
10080 INPUT/T,NO
10090 CLOSE
10100 REM'
10270 PRINT:PRINT "ｼｭｳﾘｮｳ ｼﾏｼﾀ。":PRINT
10280 CO=1:GOSUB 1970
10290 GOTO 1220
10300 REM *** INPUT
10310 PRINT "> ";:Y=CSR V
10320 FOR I=0 TO 24:A$(I)=" ":NEXT:X=2
10330 CURSOR X,Y:FOR I=X TO 23:PRINT A$(I);:NEXT
10340 CURSOR X,Y:PRINT "*";:CURSOR X,Y:PRINT A$(X);
10350 GET A$:IF A$="" GOTO 10340
10360 A=ASC(A$):IF (A<32)+(A=127) GOTO 10390
10370 A$(X)=A$:CURSOR X,Y:PRINT A$;:IF X<23 THEN X=X+1
10380 GOTO 10340
10390 IF (A=4)*(X>2) THEN X=X-1:GOTO 10340
10400 IF A=7 GOTO 10480
10410 IF (A=3)*(X<23) THEN X=X+1:GOTO 10340
10420 IF A=8 GOTO 10500
10430 IF A=13 GOTO 10510
10440 IF A=6 GOTO 10320
10450 IF A=5 THEN X=2:GOTO 10340
10460 IF A=12 GOTO 10540
10470 GOTO 10340
10480 FOR I=X TO 24:A$(I-1)=A$(I):NEXT:A$(24)=" ":IF X<>2 THEN X=X-1
10490 GOTO 10330
10500 FOR I=22 TO X STEP -1:A$(I+1)=A$(I):NEXT:A$(X)=" ":GOTO 10330
10510 A$="":A=0:FOR I=23 TO 2 STEP -1:IF A$(I)<>" " THEN A=1
10520 IF (A$(I)<>" ")+(A=1) THEN A$=A$(I)+A$
10530 NEXT:PRINT:RETURN
10540 PRINT CHR$(12);:GOTO 10330
10550 REM'** CRT
10560 B$="ｹ"+CHR$(4)
10570 READ A$:IF A$="*" THEN RETURN
10580 FOR I=1 TO LEN(A$)
10590 IF MID$(A$,I,1)="/" THEN PRINT " ":GOTO 10620
10600 PRINT MID$(A$,I,1);
10610 PRINT B$;
10620 FOR J=0 TO 120
10630 NEXT
10640 NEXT
10650 GOTO 10570
10660 REM'** END
10670 PRINT
10680 PRINT
10690 USR(&HF14)
10700 RESTORE 10710:GOSUB 10550
10710 DATA "ｸﾞｰﾑ ｵｰﾊﾞｰﾃﾞｽ。ﾓｳｲﾁﾄﾞ ｽﾙﾄｷﾊ RUN ｼﾃｸﾀﾞｻｲ。/","*"
10720 END
```

10300 〜 10540 ：必要ありません

DEUS EX MACHINA
マシン語・魔神語

Toshiyuki Nagase
長瀬敏之

基礎編「ジャンプ命令」

ジャンプ命令とは，指定した番地にプログラムの実行を移す命令です。BASIC語で言えば，「GOTO」文に相当します。

この命令の種類と動作は，リストのとおりです。

注：リストの注釈欄に「REN」という変数を使用しています。これは，その機械語命令が書き込まれているメモリの番地を意味します。

絶対番地指定と相対番地指定

ジャンプ命令には，絶対番地指定（JP）と相対番地指定（JR）の2種類があります。

●絶対番地指定

ジャンプ先を直接指定します。つまり，

JP　nn

は，「nn番地にジャンプする」，言い換えれば，「nn番地から書き込まれている命令を実行する」という意味になります。

また，nnは「0000H～FFFFH」のどの値も取りうるので，どの番地へもジャンプすることができます。

●相対番地指定

ジャンプ先を相対指定します。つまり，

JR　e

は，「この命令が書き込まれている番地にeを加えた番地へジャンプする」という意味になります。

ただし，eは「－126～＋129」の値しか取りえないので，JR命令が書き込まれている番地から－126より小さい番地と＋129より大きな番地へはジャンプできません。そのかわり，JP命令よりも命令の長さが1バイト短いし，リロケータブルなプログラムも作ることができます。

ところで，

JR　e

を機械語に直すときは，

18，e－2

としてください。

というのは，アセンブリ語ではジャンプ先がわかりやすいように，「JR命令が書き込まれている番地」を基準にして，「e」だけジャンプするように考えてありますが，機械語では，「次の命令が書き込まれている番地」を基準にしてジャンプするようになっているからです。つまり，「e」からJR命令の長さである「2」を引かなくてはならないということです。

詳細は，図1を見てください。

注：リストを見てのとおり，

JR　条件，e

の「条件」には，「ゼロ・フラグ」と「キャリー・フラグ」しか使えません。

図1　JR命令の実行

・JR命令を使用して，7010H番地へジャンプする場合

メモリの番地	機械語	アセンブリ語
7000H	18　0E	JR　10H
7002H		
⋮		
7010H		
⋮		

・左記の7000H番地のJR命令を実行すると，PCは次のように変化します。

①JR命令の前の命令が実行を終了した時……PC＝7000H

②JR命令をCPUが理解した時……PC＝7002H

③JR命令実行……PC＝7010H（＝7002H＋0EH）

・以上のように，②の段階でPCにはすでに2が加算されているので，アセンブリ語では「e」でも，機械語にするときは「e－2」としなくてはいけません。

注　PC：プログラム・カウンタ

無条件ジャンプ命令と条件付ジャンプ命令

●無条件ジャンプ命令

JP　nn

JR　e

のように，条件が付いていないジャンプ命令のことです。言うまでもなく，この命令を実行すると必ずジャンプします。

●条件付ジャンプ命令

JP　条件，nn

JR　条件，e

のように，条件が付いているジャンプ命令のことです。つまり，条件が満たされていればジャンプして，そうでないときはなにもしないで次の命令を実行します。BASIC語で言えば，「IF　条件　GOTO～」に相当します。

ここで言う「条件」は，フラグの状態のことで，次のような種類があります。

条件	フラグの状態	
NZ	ゼロ・フラグがリセット	Z＝0
Z	ゼロ・フラグが　セット	Z＝1
NC	キャリー・フラグがリセット	CY＝0
C	キャリー・フラグが　セット	CY＝1
PO	P／V・フラグがリセット	P/V＝0
PE	P／V・フラグが　セット	P/V＝1
P	サイン・フラグがリセット	S＝0
M	サイン・フラグが　セット	S＝1

その他のジャンプ命令

JP　(HL)

JP (IX)
JP (IY)

レジスタ・ペアHLや，インデックス・レジスタIX，IYの内容が，そのままジャンプ先になります。たとえば，

HL＝AB00H

のときに，

JP (HL)

を実行すると，AB00H番地へジャンプします。

●DJNZ e

一種の条件付ジャンプ命令です。レジスタBの内容を1だけ減じて，もし「B≠00H」ならば相対的にeだけジャンプして，「B＝00H」ならば，その次の命令を実行します。

この命令を他の命令で表わすと，次のようになります。

```
DEC  B
JR   NZ, e
```

ところで，同じプログラムを決められた回数だけ実行する場合は，「DJNZ」命令を使用すると便利です。たとえば，次のように使用します。

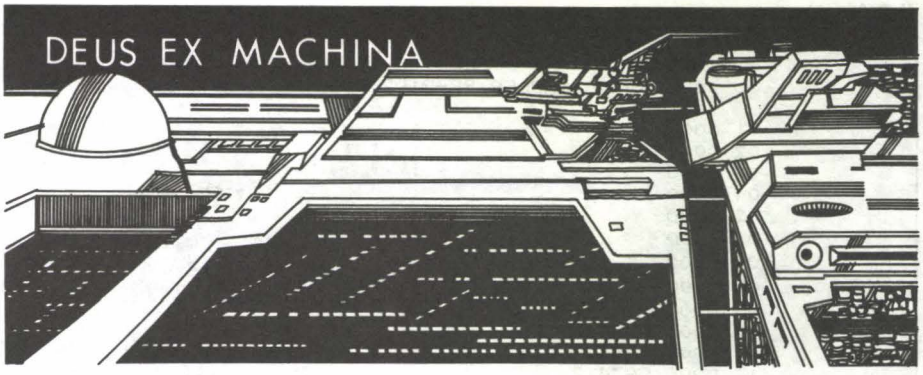
```
        ｛
        LD    B, xH
LP01 : ENT
        PUSH  BC          −①
        ┌──────┐
        │プログラムA│
        └──────┘
        POP   BC          −②
        DJNZ  LP01
        ｝
```

これは，「プログラムA」をx回実行するプログラムです（注：xH＝00H～FFH。ただし，xH＝00Hのときは，「プログラムA」を256回実行します）。

①②の「PUSH・POP」命令は，レジスタBの内容が壊れないようにするためです。しかし，「プログラムA」中でレジスタBの内容が壊れないことがわかっていれば，①②は必要ありません。また，「プログラムA」中で，

```
POP   BC
LD  B, 01H
PUSH  BC
```

を実行すると，「プログラムA」をx回実行していなくても，ループを抜け出すことができます。①②が省略されているときは，




```
LD  B, 01H
```

とするだけで，ループを抜け出すことができます。

PROGRAM 1

BASIC語の「ON～GOTO～」文に相当するプログラムです。レジスタAに，ある値を代入することによって，あらかじめ決められた番地にジャンプすることができます。

このプログラムでは，たとえば，A＝00Hならば2000H番地にジャンプ，A＝07Hならば3AFFH番地にジャンプすることができます。

さらに，「TAB01」以下のデータを変えることによって，その他の番地にもジャンプ可能なプログラムを作ることができます。

解説

10行目～ 「A＞07H」ならば，「LP01」へジャンプします。ただし「TAB01」以下のジャンプ先データの数を変えたときは，

CP「ジャンプ先データの数」

になるようにしてください。ここでは，

CP 08H

です。

12行目～ レジスタ・ペアHLに，ジャンプ先のデータを代入します。たとえば，

A＝03H

ならば，

DE＝0003H

になり，

IX＝A01CH

となるので，結局

HL＝3200H

となります。

19行目～ レジスタ・ペアHLが示す番地にジャンプします。

21行目～ ジャンプ先のテーブルです。

31行目～ 「A＞08H」ならば，ここへジャンプします。

PROGRAM 2

初項1，公差1，項数nの等差数列の和Sを求めるプログラムです。簡単に言えば，nを与えて

S＝1＋2＋3＋……＋n

の，Sを求めるプログラムです。

A00CH番地（DATA01）にnの値を代入してこのサブルーチンを実行すると，Sの値がA00DH番地（DATA02）に代入されます。たとえば，

(A00CH)＝07H

にして，このサブルーチンを実行すると，

(A00DH)＝1CH〔01H+02H+03H+04H+05H+06H+07H=1CH〕

になります。

ただし，「S＞FFH」になる場合は，オーバー・フローのため，正解は得られません。

解説

10行目～ レジスタBに，(DATA01)の値を代入する。そして，レジスタAの内容をクリアします。

13行目～ レジスタAに，レジスタBの内容を加算します。このループ内では，レジスタBの内容が，

n, n-1, n-2, ……, 3, 2, 1

というように，ひとつ減っていくことに注意してください。

16行目～ (DATA02)に，レジスタAの内容を書き込んで，リターンします。

あとがき

次回は，基礎編「コール・リターン命令，入力・出力命令」を予定しています。基礎編は次回で最終回になります。

リスト

```
    **  Z80 ASSEMBLER SP-2102 PAGE 01  **

01 0000                 ;
02 0000                 ; LIST
03 0000                 ;
04 0000                 ;          ｼﾞｬﾝﾌﾟ ﾒｲﾚｲ
05 0000                 ;
06 0000                 ;
07 0000                 ;
08 0000                 ;
09 0000                 ;
10 0000 C33412                  JP      1234H           ; GOTO 1234H
11 0003 C23412                  JP      NZ,1234H        ; IF Z=0 GOTO 1234H
12 0006 CA3412                  JP      Z,1234H         ; IF Z=1 GOTO 1234H
13 0009 D23412                  JP      NC,1234H        ; IF CY=0 GOTO 1234H
14 000C DA3412                  JP      C,1234H         ; IF CY=1 GOTO 1234H
15 000F E23412                  JP      PO,1234H        ; IF P/V=0 GOTO 1234H
16 0012 EA3412                  JP      PE,1234H        ; IF P/V=1 GOTO 1234H
17 0015 F23412                  JP      P,1234H         ; IF S=0 GOTO 1234H
18 0018 FA3412                  JP      M,1234H         ; IF S=1 GOTO 1234H
19 001B                 ;
20 001B 1810                    JR      12H             ; GOTO REN+12H
21 001D 2010                    JR      NZ,12H          ; IF Z=0 GOTO REN+12H
22 001F 2810                    JR      Z,12H           ; IF Z=1 GOTO REN+12H
23 0021 3010                    JR      NC,12H          ; IF CY=0 GOTO REN+12H
24 0023 3810                    JR      C,12H           ; IF CY=1 GOTO REN+12H
25 0025                 ;
26 0025 E9                      JP      (HL)            ; GOTO (HL)
27 0026 DDE9                    JP      (IX)            ; GOTO (IX)
28 0028 FDE9                    JP      (IY)            ; GOTO (IY)
29 002A                 ;
30 002A 10EC                    DJNZ    -12H            ; B=B-1
31 002C                 ;                                 IF B>0 GOTO REN-12H
32 002C                 ;
33 002C                 ;
34 002C                 ;
35 002C                         END
```

PROGRAM 1

```
    **  Z80 ASSEMBLER SP-2102 PAGE 01  **

01 0000                 ;
02 0000                 ; PROGRAM 1
03 0000                 ;
04 0000                 ;          ON A GOTO -
05 0000                 ;
06 0000                 ;
07 0000                 ;
08 0000                         REL     A000H           ; ORG $A000
09 A000                 ;
10 A000 FE08                    CP      08H             ; IF A<08H ELSE CY=0
11 A002 3022                    JR      NC,LP01         ; IF CY=0 GOTO LP01
12 A004 5F                      LD      E,A             ; E=A
13 A005 1600                    LD      D,00H           ; D=00H
14 A007 DD2116A0                LD      IX,TAB01        ; IX=TAB01
15 A00B DD19                    ADD     IX,DE           ; IX=IX+DE
16 A00D DD19                    ADD     IX,DE           ; IX=IX+DE
17 A00F DD6E00                  LD      L,(IX+00H)      ; L=(IX+00H)
18 A012 DD6601                  LD      H,(IX+01H)      ; H=(IX+01H)
19 A015 E9                      JP      (HL)            ; GOTO (HL)
20 A016                 ;
21 A016                 TAB01:  ENT
22 A016 0020                    DEFW    2000H           ; A=00H
23 A018 0021                    DEFW    2100H           ; A=01H
24 A01A 0023                    DEFW    2300H           ; A=02H
25 A01C 0032                    DEFW    3200H           ; A=03H
26 A01E 003A                    DEFW    3A00H           ; A=04H
27 A020 183A                    DEFW    3A18H           ; A=05H
28 A022 583A                    DEFW    3A58H           ; A=06H
29 A024 FF3A                    DEFW    3AFFH           ; A=07H
30 A026                 ;
31 A026                 LP01:   ENT
32 A026                 ;
33 A026                 ;
34 A026                 ;
35 A026                         END
```

PROGRAM 2

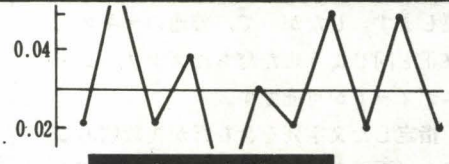
```
    **  Z80 ASSEMBLER SP-2102 PAGE 01  **


01 0000                 ;
02 0000                 ; PROGRAM 2
03 0000                 ;
04 0000                 ;          (DAT02)=01H+02H+...+(DAT01)
05 0000                 ;
06 0000                 ;
07 0000                 ;
08 0000                         REL   A000H             ; ORG $A000
09 A000                 ;
10 A000 3A0CA0                  LD    A,(DAT01)         ; A=(DAT01)
11 A003 47                      LD    B,A               ; B=A=(DAT01)
12 A004 AF                      XOR   A                 ; A=00H
13 A005                 LP01:   ENT
14 A005 80                      ADD   A,B               ; A=A+B
15 A006 10FD                    DJNZ  LP01              ; B=B-1:IF B>0 GOTO LP01
16 A008 320DA0                  LD    (DAT02),A         ; (DAT02)=A
17 A00B                 ;
18 A00B C9                      RET                     ; RETURN
19 A00C                 ;
20 A00C                 ;
21 A00C                 DAT01:  ENT
22 A00C 00                      DEFB  00H
23 A00D                 DAT02:  ENT
24 A00D 00                      DEFB  00H
25 A00E                 ;
26 A00E                 ;
27 A00E                 ;
28 A00E                         END
```


実績/教材/講師陣 ナンバー1を誇る！

文部省認定 行政管理庁指定 統計解析通信講座

●昭和59年度第20期生受付中！ 統計官(主事)資格が取れる。

●唯一の文部省認定/行政管理庁指定/通信講座
●執筆・指導＝上智大教授・理博　斎藤金一郎
前東京大教授・理博　奥野　忠一
千葉大教授・理博　浅井　晃
慶応大講師・工博　芳賀　敏郎



統計資料無料呈
●ハガキか電話で直接下記までどうぞ！

★講座の特色★

- 統計学を初めて学ぶ人、さらに理解を深めたい人に、今こそ、統計を自分のものにするタイムリーな講座。
- 膨大な理論と広範な応用領域をもつ現代統計学の集大成決定版。
- 統計の基礎から応用まで、豊富な実例によるシステム教材で短期指導。
- 提出レポートに個別添削指導を実施
- 修了者に行政管理庁指定《統計官・統計主事》資格証書を交付。

★主要内容★

第1単元＝統計とは何か
第2単元＝集団構造の記述
第3単元＝母集団と標本
第4単元＝推定と検定
第5単元＝回帰と相関
第6単元＝統計調査
第7単元＝標本調査法
第8単元＝品質管理と実験計画

(財)実務教育研究所 公開講座統計部
160 東京都新宿区大京町4の383
☎東京03(357)8153

マシン語に挑戦しよう!

BASIC文字列サーチ・ツール

Yoshigi Ogasawara
小笠原吉義

はじめに

マシン語の文字コード列を探すプログラムを本誌'83年6月号で紹介しました。今回，BASICテキスト内の任意の文字列を探すプログラムを作りましたのでご紹介します。BASICプログラムの修正や分析に役立つかと思います。

1.特長

BASICテキスト内の文字列をサーチする場合，BASICプログラムは中間コードに変換されているため，普通の16進コードのサーチとは違って中間コードをもとのステートメントに戻してからサーチする必要があります。そこで，この変換にはBASICインタプリタのルーチンを利用しました。操作性が良くなるように特に次の点を工夫してあります。

(1) 指定した文字列を含むBASICプログラムの行を探し出すと，その行を1行分だけ表示してBASICインタプリタに制御を戻します。したがって，普通のテキスト修正と同じようにただちにスクリーン・エディットが可能です。

(2) 指定した文字列を含む行が複数個ある場合，F10 キーを押すと1行ごとに順次サーチを繰り返して表示します。

(3) 新しい文字列を指定したい場合は，F9 キーを押すとサーチ文字列の入力モードになります。

(4) 空白(SPACE)の文字については，サーチの対象からはずして無視するようにしました。BASICプログラムの実行時も無視されるからです。たとえば，

GOTO55 と GOTO␣55

は同じ文字列として扱います。

(5) BASICインタプリタのサブルーチンを利用していますが，BASICインタプリタの内部をパッチ(変更)してはいないので，BASICプログラムの実行には影響を与えません。

2.プログラムの入力

テープBASIC(MZ-1Z001)の場合はリスト1，またディスクBASIC(MZ-2Z001)はリスト2を参照して，モニタのMコマンドでF800番地からF9FF番地までを入力してください。リロケータブルではありません。

正しく入力できたら，モニタのSコマンドでスタートアドレスF800番地，エンドアドレスF9FF番地として，テープにSAVEしてください。J-ADR.$に対して，すぐにCR

図1 (使用例)

```
** MONITOR MZ-1Z001M **                          ┐
                                                  │
 BASIC interpreter  MZ-1Z001 V1.0C                │ BASIC
                                                  │ IPL
 Copyright 1982 by SHARP Corp.                    │
                                                  ┘
 43500 Bytes
Ready
LOAD                                              ┐
FOUND "ﾁﾃｲ ｻｲﾀﾞｲ ﾉ ｻｸｾﾝ"                          │ BASICプログラム
LOADING "ﾁﾃｲ ｻｲﾀﾞｲ ﾉ ｻｸｾﾝ"                        │ ロード
Ready                                             ┘
LIMIT$F7FF                                        ┐ マシン語エリア確保
Ready                                             ┘
MON                                               ┐
*L                                                │ BASIC文字列サーチ・ツール
FILE NAME:                                        │ ロード
LOADING BSCH-CMT(*F800)                           ┘
*J                                                ┐ ツール起動
J-ADR.$F800                                       ┘
┌──────(MEMO)──────┐                            ┐
| NEW SEARCH --PUSH(F9)  |                       │ コメント
| NEXT SEARCH--PUSH(F10) |                       ┘
└──────────────────┘

*SEARCH STRINGS((16CHR.)?                         ┐ サーチ文字列"GRAPH"
GRAPH                                             ┘
*SEARCH START LINE NO.(Def:0)?                    ┐ 先頭からサーチ開始(CR)
 30 CONSOLE0,24,C40,N:LIMIT$C000:CONSOL           ┘
EGN:GRAPHC:GRAPHI2,0123
                                                  ←ファンクション・キー F10
 114 GRAPHI2
                                                  ←ファンクション・キー F10
      :                                           :
      :                                           :
 16420 GETA$:IFA$="S"THENGRAPHI1,C,I2,
I3,C:RETURN                                       ←ファンクション・キー F10
********(SEARCH END)********                      ┐
┌──────(MEMO)──────┐                            │ サーチ終了メッセージ
| NEW SEARCH --PUSH(F9)  |                       │ (新サーチの場合
| NEXT SEARCH--PUSH(F10) |                       │ (ファンクション・キー F9)
└──────────────────┘                            ┘

Ready
```

を入力してジャンプアドレスを省略してもいいのですが，Oh！MZチェックサム・プログラム（本誌'83．6月号）で入力ミスがないことが確認できたら，ジャンプアドレスをF800番地としてテープにSAVEしておいたほうが便利です。

ディスクBASICの場合は，SAVEしたテープから〝Filing CMT〟のユーティリティを使用してディスクに登録してください。

3.プログラムの起動

(1) テープBASICのとき

①サーチ対象のBASICプログラムをあらかじめLOADしておく

②LIMIT ＄F7FF CR

③MON CR

④SAVEしたプログラムテープをセット

⑤＊L CR

(⑥＊J CR)

(⑦J-ADR.＄F800 CR)

⑥と⑦はSAVEのときにジャンプアドレスを指定していれば不要です。

(2) ディスクBASICのとき

①サーチ対象のBASICプログラムをLOAD

②LIMIT ＄F7FF CR

③LOAD〝このツールのファイル名〟CR

④MON CR

⑤＊J CR

⑥J-ADR.＄ F800 CR

これで，サーチするプログラムが起動されます。RUNコマンドでこのプログラムを起動しないでください。

4.プログラムの使い方

①最初にサーチ文字列の入力モードになります。16文字以内の文字列を指定してください。省略すると全行出力になります。

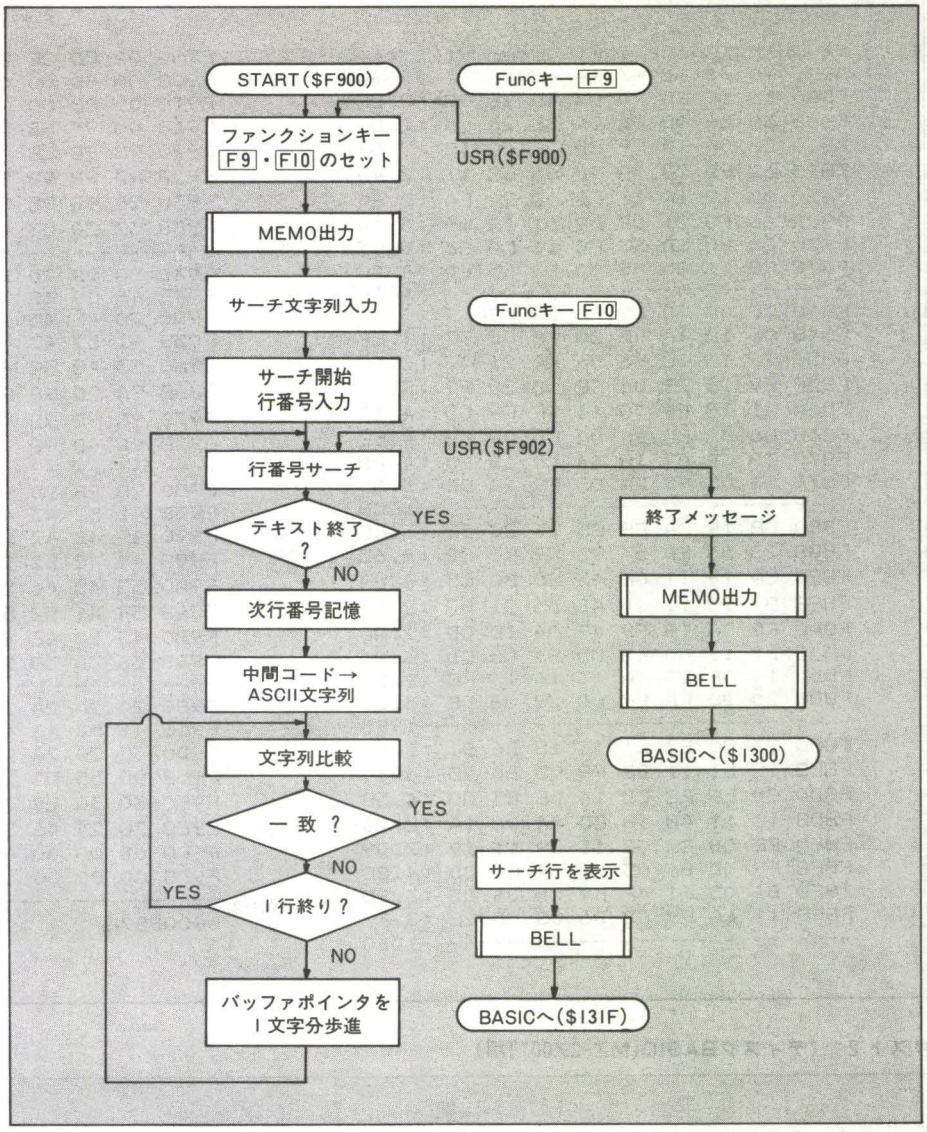
②次に，サーチを開始する行番号を入力してください。省略するとテキストの先頭からサーチします。

③指定した文字列と同じ文字列（空白は無視）があると，その行を表示します。サーチを続行したい場合にはF10キーを押してください。

④新しいサーチ文字列を指定したいときは，F9キーを押してください。

図1に使用例を示します。

図2　フローチャート



5.コメント

(1) このプログラムを実行すると，

DEF KEY(9)＝USR(＄F800) CR

DEF KEY(10)＝USR(＄F802) CR

を投入した形になります。KLISTコマンドで確認してみてください。ファンクションキーの定義領域は1200番地から12A0番地までで，ここを更新しています。詳しくはリスト3のアセンブルリストを参考にしてください。

(2) サーチ文字列を省略してすぐにCRとすると，1行ずつ全部の行をF10キーで表示させることができます。BASICプログラムを入力し終わって，入力ミスがないかどうかをチェックするとき，この使い方をすると便利です。

6.プログラムの説明

●概略フローチャートを図2に示します。

●CALL BIN（CMTのとき＄18A7，ディスクのとき＄1A23）は，10進ASCII文字を2バイトの2進数に変換するBASICインタプリタのサブルーチンです。

●CALL LSR（CMTのとき＄1988，ディスクのとき＄1B04）は，HLレジスタに格納されている行番号をサーチするサブルーチンです。

●CALL BCNT（CMTのとき＄19CA，ディスクのとき＄1B46）は，行の長さ（バイト数）をカウントしてBCレジスタにセットするサブルーチン。

●他のサブルーチンはBASIC/MONITORマニュアルを参考にしてください。

リスト1　（テープBASIC〔MZ-1Z001〕用）

```
** Oh!MZ Check sum System V1.1 **
ADR. +0 +1 +2 +3 +4 +5 +6 +7 :sum.
F800 18 17 ED 7B 0A 4D 3E 02 :2.2E
F808 CD C6 08 06 0A 3E 20 CD :2.D6
F810 C6 08 10 F9 CD 2E 0A 18 :2.F4
F818 62 ED 7B 0A 4D 06 08 11 :2.40
F820 00 12 1A 13 FE 0D 20 FA :2.64
F828 10 F8 21 D9 F9 01 18 00 :3.14
F830 ED B0 ED 53 F0 11 CD E6 :5.91
F838 F8 11 85 F9 CD B6 05 CD :4.DC
--------------------------------1A1D
F840 2E 0A CD D9 F8 21 07 F9 :3.F7
F848 06 10 1A FE 20 28 02 77 :1.EF
F850 23 13 10 F6 06 FF 21 07 :2.69
F858 F9 7E 23 04 FE 0D 20 F9 :3.C2
F860 21 18 F9 70 11 9F F9 CD :4.18
F868 B6 05 CD 2E 0A CD D9 F8 :4.5E
F870 EB CD A7 18 64 F8 1B ED :4.DB
F878 53 05 F9 2A 05 F9 23 CD :3.69
--------------------------------1CCB
F880 88 19 CA F8 E5 23 23 5E :3.EC
F888 23 56 ED 53 05 F9 E1 CD :4.65
F890 CA 19 11 0E 4E ED B0 CD :3.BA
F898 87 15 11 0D 4D D5 21 07 :2.04
F8A0 F9 3A 18 F9 47 04 10 0D :2.AC
F8A8 D1 11 0D 4D CD B6 05 CD :3.91
F8B0 14 0F C3 0A 13 1A 13 FE :2.2E
F8B8 20 28 FA FE 0D 28 08 BE :3.3B
--------------------------------19B5
F8C0 23 28 E3 D1 13 18 D6 D1 :3.D1
F8C8 18 B1 11 BE F9 CD B6 05 :4.19
F8D0 CD E6 F8 CD 14 0F C3 00 :4.5E
F8D8 13 11 AB 10 CD A4 06 1A :2.70
F8E0 FE 0B CA 00 13 C9 11 19 :2.D9
F8E8 F9 CD B6 05 11 34 F9 CD :4.8C
F8F0 B6 05 11 4F F9 CD B6 05 :3.9C
F8F8 11 6A F9 CD B6 05 CD 2E :3.F7
--------------------------------1DB0
F900 0A CD 2E 0A C9 00 00 00 :1.D8
F908 00 00 00 00 00 00 00 00 :0.00
F910 00 00 00 00 00 00 00 0D :0.0D
F918 00 97 9B 9B 9B 9B 9B 9B :4.39
F920 9B 9B 28 4D 45 4D 4F 29 :2.B5
F928 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
F930 9B 9B 95 0D 9A 20 4E 45 :3.25
F938 57 20 53 45 41 52 43 48 :2.2D
--------------------------------12FD
F940 20 2D 2D 50 55 53 48 3C :1.F6
F948 46 39 3E 20 20 9A 0D 9A :2.3E
F950 20 4E 45 58 54 20 53 45 :2.17
F958 41 52 43 48 2D 2D 50 55 :2.1D
F960 53 48 3C 46 31 30 3E 20 :1.DC
F968 9A 0D 98 9B 9B 9B 9B 9B :4.46
F970 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
F978 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
--------------------------------183A
F980 9B 9B 9B 96 0D 2A 53 45 :3.36
F988 41 52 43 48 20 53 54 52 :2.37
F990 49 4E 47 53 28 3C 31 36 :1.FC
F998 43 48 52 2E 29 3F 0D 2A :1.AA
F9A0 53 45 41 52 43 48 20 53 :2.29
F9A8 54 41 52 54 20 4C 49 4E :2.3E
F9B0 45 20 4E 4F 2E 28 44 65 :2.01
F9B8 66 3A 30 29 3F 0D 2A 2A :1.99
--------------------------------1114
F9C0 2A 2A 2A 2A 2A 3C 53 45 :1.A6
F9C8 41 52 43 48 20 45 4E 44 :2.15
F9D0 3E 2A 2A 2A 2A 2A 2A 2A :1.64
F9D8 0D 55 53 52 28 24 46 38 :1.D1
F9E0 30 30 29 7F 0D 55 53 52 :2.0F
F9E8 28 24 46 38 30 32 29 7F :1.D4
F9F0 0D 00 00 00 00 00 00 00 :0.0D
F9F8 00 00 00 00 00 00 00 00 :0.00
--------------------------------0AE0
0000B578
```

リスト2　（ディスクBASIC〔MZ-2Z001〕用）

```
** Oh!MZ Check sum System V1.1 **
ADR. +0 +1 +2 +3 +4 +5 +6 +7 :sum.
F800 18 17 ED 7B 0A 6E 3E 02 :2.4F
F808 CD C6 08 06 0A 3E 20 CD :2.D6
F810 C6 08 10 F9 CD 2E 0A 18 :2.F4
F818 62 ED 7B 0A 6E 06 08 11 :2.61
F820 00 12 1A 13 FE 0D 20 FA :2.64
F828 10 F8 21 D9 F9 01 18 00 :3.14
F830 ED B0 ED 53 F0 11 CD E6 :5.91
F838 F8 11 85 F9 CD B6 05 CD :4.DC
--------------------------------1A5F
F840 2E 0A CD D9 F8 21 07 F9 :3.F7
F848 06 10 1A FE 20 28 02 77 :1.EF
F850 23 13 10 F6 06 FF 21 07 :2.69
F858 F9 7E 23 04 FE 0D 20 F9 :3.C2
F860 21 18 F9 70 11 9F F9 CD :4.18
F868 B6 05 CD 2E 0A CD D9 F8 :4.5E
F870 EB CD 23 1A 64 F8 1B ED :4.59
F878 53 05 F9 2A 05 F9 23 CD :3.69
--------------------------------1C49
F880 04 1B CA F8 E5 23 23 5E :3.6A
F888 23 56 ED 53 05 F9 E1 CD :4.65
F890 46 1B 11 0E 6F ED B0 CD :3.59
F898 68 16 11 0D 6E D5 21 07 :2.07
F8A0 F9 3A 18 F9 47 04 10 0D :2.AC
F8A8 D1 11 0D 6E CD B6 05 CD :3.B2
F8B0 14 0F C3 1F 13 1A 13 FE :2.43
F8B8 20 28 FA FE 0D 28 08 BE :3.3B
--------------------------------190B
F8C0 23 28 E3 D1 13 18 D6 D1 :3.D1
F8C8 18 B1 11 BE F9 CD B6 05 :4.19
F8D0 CD E6 F8 CD 14 0F C3 00 :4.5E
F8D8 13 11 AB 10 CD A4 06 1A :2.70
F8E0 FE 0B CA 00 13 C9 11 19 :2.D9
F8E8 F9 CD B6 05 11 34 F9 CD :4.8C
F8F0 B6 05 11 4F F9 CD B6 05 :3.9C
F8F8 11 6A F9 CD B6 05 CD 2E :3.F7
--------------------------------1DB0
F900 0A CD 2E 0A C9 00 00 00 :1.D8
F908 00 00 00 00 00 00 00 00 :0.00
F910 00 00 00 00 00 00 00 0D :0.0D
F918 00 97 9B 9B 9B 9B 9B 9B :4.39
F920 9B 9B 28 4D 45 4D 4F 29 :2.B5
F928 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
F930 9B 9B 95 0D 9A 20 4E 45 :3.25
F938 57 20 53 45 41 52 43 48 :2.2D
--------------------------------12FD
F940 20 2D 2D 50 55 53 48 3C :1.F6
F948 46 39 3E 20 20 9A 0D 9A :2.3E
F950 20 4E 45 58 54 20 53 45 :2.17
F958 41 52 43 48 2D 2D 50 55 :2.1D
F960 53 48 3C 46 31 30 3E 20 :1.DC
F968 9A 0D 98 9B 9B 9B 9B 9B :4.46
F970 9B 9B 9B 9B 9B 9B 9B 9B :4.D8
```

```
F97B 9B 9B 9B 9B 9B 9B 9B 9B :4.DB
--------------------------------183A
F980 9B 9B 9B 96 0D 2A 53 45 :3.36
F988 41 52 43 48 20 53 54 52 :2.37
F990 49 4E 47 53 28 3C 31 36 :1.FC
F998 43 48 52 2E 29 3F 0D 2A :1.AA
F9A0 53 45 41 52 43 48 20 53 :2.29
F9A8 54 41 52 54 20 4C 49 4E :2.3E
F9B0 45 20 4E 4F 2E 28 44 65 :2.01
F9B8 66 3A 30 29 3F 0D 2A 2A :1.99
--------------------------------1114
```

```
F9C0 2A 2A 2A 2A 2A 3C 53 45 :1.A6
F9C8 41 52 43 48 20 45 4E 44 :2.15
F9D0 3E 2A 2A 2A 2A 2A 2A 2A :1.64
F9D8 0D 55 53 52 28 24 46 38 :1.D1
F9E0 30 30 29 7F 0D 55 53 52 :2.0F
F9E8 28 24 46 38 30 32 29 7F :1.D4
F9F0 0D 00 00 00 00 00 00 00 :0.0D
F9F8 00 00 00 00 00 00 00 00 :0.00
--------------------------------0AE0

0000B48E
```

リスト3 （アセンブル・リスト）

```
  **  Z80 ASSEMBLER 2Z-004C  <BSCH-CMT>  PAGE 01          01.16.84

01 0000                 ;**************************
02 0000                 ;* BASIC-TEXT-SEARCH-TOOL *
03 0000                 ;*  (MZ-1Z001 V1.0C)      *
04 0000                 ;*  1984.1.15 BY Y.O      *
05 0000                 ;**************************
06 0000                 ;
07 F800                         ORG     F800H
08 10AB  P              BUFER:  EQU     10ABH
09 08C6  P              PRNT:   EQU     08C6H
10 06A4  P              GETL:   EQU     06A4H
11 0A2E  P              LETNL:  EQU     0A2EH
12 05B6  P              NLMSG:  EQU     05B6H
13 0F14  P              BELL:   EQU     0F14H
14 130A  P              BAS2:   EQU     130AH           ;FD=131FH
15 18A7  P              BIN:    EQU     18A7H           ;FD=1A23H
16 1988  P              LSR:    EQU     1988H           ;FD=1B04H
17 4E10  P              BBUF:   EQU     4E10H           ;FD=6F10H
18 4D0D  P              CBUF:   EQU     4D0DH           ;FD=6E0DH
19 1587  P              TEXT:   EQU     1587H           ;FD=1668H
20 19CA  P              BCNT:   EQU     19CAH           ;FD=1B46H
21 4D0A  P              SPA:    EQU     4D0AH           ;FD=6E0AH
22 F800                 ;
23 F800 1817            USR1:   JR      START
24 F802 ED7B0A4D        USR2:   LD      SP,(SPA)
25 F806 3E02                    LD      A,02H
26 F808 CDC608                  CALL    PRNT
27 F80B 060A                    LD      B,10
28 F80D 3E20            U21:    LD      A,20H
29 F80F CDC608                  CALL    PRNT
30 F812 10F9                    DJNZ    U21
31 F814 CD2E0A                  CALL    LETNL
32 F817 1862                    JR      NEXT
33 F819                 ;
34 F819 ED7B0A4D        START:  LD      SP,(SPA)
35 F81D 0608                    LD      B,8
36 F81F 110012                  LD      DE,1200H
37 F822 1A              D1:     LD      A,(DE)
38 F823 13                      INC     DE
39 F824 FE0D                    CP      0DH
40 F826 20FA                    JR      NZ,D1
41 F828 10F8                    DJNZ    D1
42 F82A 21D9F9                  LD      HL,DEFK
43 F82D 011800                  LD      BC,24
44 F830 EDB0                    LDIR
45 F832 ED53F011                LD      (11F0H),DE
46 F836                 ;
47 F836 CDE6F8                  CALL    MEMO
48 F839 1185F9                  LD      DE,MSG5
49 F83C CDB605                  CALL    NLMSG
50 F83F CD2E0A                  CALL    LETNL
51 F842 CDD9F8                  CALL    GETLB
52 F845 2107F9                  LD      HL,STRING
53 F848 0610                    LD      B,16
54 F84A 1A              I1:     LD      A,(DE)
55 F84B FE20                    CP      20H
56 F84D 2802                    JR      Z,I2
57 F84F 77                      LD      (HL),A
58 F850 23                      INC     HL
59 F851 13              I2:     INC     DE
60 F852 10F6                    DJNZ    I1
```

```
  **  Z80 ASSEMBLER 2Z-004C  <BSCH-CMT>  PAGE 02          01.16.84

01 F854                 ;
02 F854 06FF                    LD      B,FFH
03 F856 2107F9                  LD      HL,STRING
04 F859 7E              I3:     LD      A,(HL)
05 F85A 23                      INC     HL
06 F85B 04                      INC     B
07 F85C FE0D                    CP      0DH
08 F85E 20F9                    JR      NZ,I3
09 F860 2118F9                  LD      HL,STN
10 F863 70                      LD      (HL),B
11 F864                 ;
12 F864 119FF9          I4:     LD      DE,MSG6
13 F867 CDB605                  CALL    NLMSG
14 F86A CD2E0A                  CALL    LETNL
15 F86D CDD9F8                  CALL    GETLB
16 F870 EB                      EX      DE,HL
17 F871 CDA718                  CALL    BIN
18 F874 64F8                    DEFW    I4
19 F876 1B                      DEC     DE
20 F877 ED5305F9                LD      (LN),DE
21 F87B                 ;
22 F87B 2A05F9          NEXT:   LD      HL,(LN)
23 F87E 23                      INC     HL
24 F87F CD8819                  CALL    LSR
25 F882 CAF8                    DEFW    SEND
26 F884 E5                      PUSH    HL
27 F885 23                      INC     HL
28 F886 23                      INC     HL
29 F887 5E                      LD      E,(HL)
30 F888 23                      INC     HL
31 F889 56                      LD      D,(HL)
32 F88A ED5305F9                LD      (LN),DE
33 F88E E1                      POP     HL
34 F88F CDCA19                  CALL    BCNT
35 F892 110E4E                  LD      DE,BBUF-2
36 F895 EDB0                    LDIR
37 F897 CD8715                  CALL    TEXT
38 F89A                 ;
39 F89A                 ;****CHR.CHECK
40 F89A 110D4D                  LD      DE,CBUF
41 F89D D5              E1:     PUSH    DE
42 F89E 2107F9                  LD      HL,STRING
43 F8A1 3A18F9                  LD      A,(STN)
44 F8A4 47                      LD      B,A
45 F8A5 04                      INC     B
46 F8A6 100D            E2:     DJNZ    E3
47 F8A8 D1                      POP     DE
48 F8A9 110D4D                  LD      DE,CBUF
49 F8AC CDB605                  CALL    NLMSG
50 F8AF CD140F                  CALL    BELL
51 F8B2 C30A13                  JP      BAS2
52 F8B5                 ;
53 F8B5 1A              E3:     LD      A,(DE)
54 F8B6 13                      INC     DE
55 F8B7 FE20                    CP      20H
56 F8B9 28FA                    JR      Z,E3
57 F8BB FE0D                    CP      0DH
58 F8BD 2808                    JR      Z,NL
59 F8BF BE                      CP      (HL)
60 F8C0 23                      INC     HL
```

```
  **  Z80 ASSEMBLER 2Z-004C  <BSCH-CMT>  PAGE 03          01.16.84

01 F8C1 28E3                    JR      Z,E2
02 F8C3 D1                      POP     DE
03 F8C4 13                      INC     DE
04 F8C5 18D6                    JR      E1
05 F8C7 D1              NL:     POP     DE
06 F8C8 18B1                    JR      NEXT
07 F8CA                 ;
08 F8CA                 ;
09 F8CA 11BEF9          SEND:   LD      DE,MSG7
10 F8CD CDB605                  CALL    NLMSG
11 F8D0 CDE6F8                  CALL    MEMO
12 F8D3 CD140F                  CALL    BELL
13 F8D6 C30013                  JP      1300H
14 F8D9                 ;
15 F8D9                 ;*** GET LINE SUB.***
16 F8D9 11AB10          GETLB:  LD      DE,BUFER
17 F8DC CDA406                  CALL    GETL
18 F8DF 1A                      LD      A,(DE)
19 F8E0 FE0B                    CP      0BH
20 F8E2 CA0013                  JP      Z,1300H
21 F8E5 C9                      RET
22 F8E6                 ;
```

MZファンは幸せモンだ。

3拍子揃えて発売中

感動！のリアルタイムゲーム

Building Hopper

For MZ 700, 1200
¥3500

高さ120mのビルを登れ！行手をはばむモンスター達はジャンプキックでやっつけろ‼MZファン注目の的のリアルタイムゲームは今、絶好調でバク進中です。

難解！のアドベンチャーゲーム

WONDER HOUSE

For MZ 700 & 1200
MZ 2200　FM 7,8
PC 6001, 8001, 8001mkII
8801, 9801
¥3500　x1

ある日届いた手紙が一通、魔女がすむ古い屋敷にダイヤモンドが隠されていると言う。気がつけばあなたは、その屋敷の前に立っていた！超難解アドベンチャー。

本格！のマージャンゲーム

COMPUTER MAHJONG

For MZ 700, 1200
¥3500

コンピューターと一騎うちの2人打マージャンゲーム。自動役判定は6倍役満までを判定、BGMも流れるキャラクタグラフィックによる本格志向のマージャンです。

近日予定！

X1ファンも幸せになる

Mr.Pac!

リアルタイムゲーム。単純明解さの中に隠れた深い内容と数知れないゲームストーリィを最後まで見る事ができるか!?

片山まさゆきの COMPUTER MAHJONG

あの片山まさゆき氏のキャラクターがついにX1に登場！勿論プログラムも負けず劣らずのオールマシン語です。豊臣くんをかっぱぎ状態にしてあげて下さい！

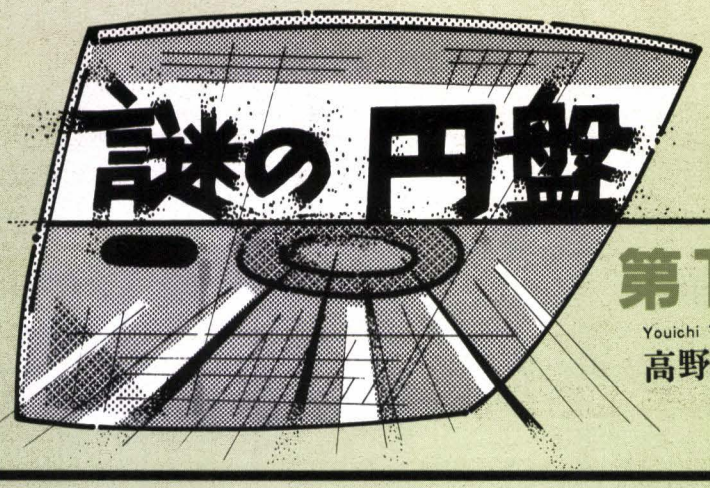
タスクフォーツ高知
〒780 高知市朝倉丙434-14
☎0888(40)1207

X1用商品とカタログの製作が遅れております。申し訳ありませんが今しばらくお待ち下さい。
通販お申込の方は商品名・機種名を明記の上代金＋送料300円を当社までお送り下さい。


```
23 F8E6                 ;***MEMO SUB.***
24 F8E6 1119F9          MEMO:   LD      DE,MSG1
25 F8E9 CDB605                  CALL    NLMSG
26 F8EC 1134F9                  LD      DE,MSG2
27 F8EF CDB605                  CALL    NLMSG
28 F8F2 114FF9                  LD      DE,MSG3
29 F8F5 CDB605                  CALL    NLMSG
30 F8F8 116AF9                  LD      DE,MSG4
31 F8FB CDB605                  CALL    NLMSG
32 F8FE CD2E0A                  CALL    LETNL
33 F901 CD2E0A                  CALL    LETNL
34 F904 C9                      RET
35 F905                 ;
36 F905 0000            LN:     DEFW    0
37 F907                 STRING: DEFS    16
38 F917 0D                      DEFB    0DH
39 F918 00              STN:    DEFB    0
40 F919 979B9B9B        MSG1:   DEFM    '┌──────────<MEMO>──────────┐'
41 F91D 9B9B9B9B
42 F921 9B284D45
43 F925 4D4F299B
44 F929 9B9B9B9B
45 F92D 9B9B9B9B
46 F931 9B95
47 F933 0D                      DEFB    0DH
48 F934 9A204E45        MSG2:   DEFM    '| NEW SEARCH --PUSH<F9>  |'
49 F938 57205345
50 F93C 41524348
51 F940 202D2D50
52 F944 5553483C
53 F948 46393E20
54 F94C 209A
55 F94E 0D                      DEFB    0DH
56 F94F 9A204E45        MSG3:   DEFM    '| NEXT SEARCH--PUSH<F10> |'
57 F953 58542053
58 F957 45415243
59 F95B 482D2D50
60 F95F 5553483C
```

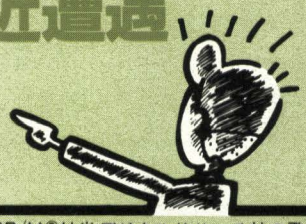
```
   **  Z80 ASSEMBLER 2Z-004C  <BSCH-CMT>  PAGE 04              01.16.84

01 F963 4631303E
02 F967 209A
03 F969 0D                      DEFB    0DH
04 F96A 989B9B9B        MSG4:   DEFM    '└──────────────────────────┘'
05 F96E 9B9B9B9B
06 F972 9B9B9B9B
07 F976 9B9B9B9B
08 F97A 9B9B9B9B
09 F97E 9B9B9B9B
10 F982 9B96
11 F984 0D                      DEFB    0DH
12 F985 2A534541        MSG5:   DEFM    '*SEARCH STRINGS(<16CHR.)?'
13 F989 52434820
14 F98D 53545249
15 F991 4E475328
16 F995 3C313643
17 F999 48522E29
18 F99D 3F
19 F99E 0D                      DEFB    0DH
20 F99F 2A534541        MSG6:   DEFM    '*SEARCH START LINE NO.(Def:0)?'
21 F9A3 52434820
22 F9A7 53544152
23 F9AB 54204C49
24 F9AF 4E45204E
25 F9B3 4F2E2844
26 F9B7 65663A30
27 F9BB 293F
28 F9BD 0D                      DEFB    0DH
29 F9BE 2A2A2A2A        MSG7:   DEFM    '********<SEARCH END>********'
30 F9C2 2A2A2A3C
31 F9C6 53454152
32 F9CA 43482045
33 F9CE 4E443E2A
34 F9D2 2A2A2A2A
35 F9D6 2A2A
36 F9D8 0D                      DEFB    0DH
37 F9D9 55535228        DEFK:   DEFM    'USR(#F800)↵'
38 F9DD 24463830
39 F9E1 30297F
40 F9E4 0D                      DEFB    0DH
41 F9E5 55535228                DEFM    'USR(#F802)↵'
42 F9E9 24463830
43 F9ED 32297F
44 F9F0 0D                      DEFB    0DH
45 F9F1                         END     USR1
```

```
   **  Z80 ASSEMBLER 2Z-004C  <BSCH-CMT>  PAGE 05              01.16.84

BAS2    130A    BBUF    4E10    BCNT    19CA    BELL    0F14    BIN     18A7
BUFER   10AB    CBUF    4D0D    D1      F822    DEFK    F9D9    E1      F89D
E2      F8A6    E3      F8B5    GETL    06A4    GETLB   F8D9    I1      F84A
I2      F851    I3      F859    I4      F864    LETNL   0A2E    LN      F905
LSR     1908    MEMO    F8E6    MSG1    F919    MSG2    F934    MSG3    F94F
MSG4    F96A    MSG5    F985    MSG6    F99F    MSG7    F9BE    NEXT    F87B
NL      F8C7    NLMSG   05B6    PRNT    08C6    SEND    F8CA    SPA     4D0A
START   F819    STN     F918    STRING  F907    TEXT    1587    U21     F80D
USR1    F800    USR2    F802
```



謎の円盤 CP/M® 第1種接近遭遇

Youichi Takano
高野庸一



CP/M®は米デジタルリサーチ社の登録商標です。

昨年のことですが，X1用のCP/Mが，シャープより16,800円で発売されましたね。一般にCP/Mといえば30,000円以上，だいたいの場合60,000円台であることを考えると，これは実に快挙だといえます。しかも，WORDMASTERまで付属しています。BIOSの作成，FORMATなどのマシン固有のツール，ディスクへの書き込み，流通などの手間を考えると，16,800円という値段は，ギリギリのところではないでしょうか。シャープの英断には，拍手をおしまないところです。

というわけで，誰にでも手が出るX1用CP/Mを中心に，今月から12回の予定で連載を始めます。ただし，MZ-80B，2000/2200のCP/Mユーザーにも役に立つプログラムにしますので，安心してください。では始めましょう。

第一章 CP/Mのおさらい

最初からおさらいというのは，おかしいかもしれませんが，Oh！MZのバックナンバー（83年11，12，84年2月号）に，CP/Mの解説はありますし，またよくできたCP/Mの入門書，解説書も数多くありますので，私はCP/Mの解説を1から始めるつもりはありません。必要なつど，少しずつ説明はします。この連載の目的は，入門ではなく，いかにしてCP/Mを使うかだからです。というわけで，CP/Mを知らない人は，まず最低限度の勉強をすませておきましょう。その場合のお勧め品は，

1) 入門，実習，応用CP/M（3部作），村瀬康治著，(株)アスキー，1982
2) 標準CP/Mハンドブック，Roday Zaks著　村瀬康治訳，(株)アスキー，1982

ですが，どちらも高価な本ですから，覚悟を決めてください。せっかく手に入れたCP/Mを生かせるかどうかの瀬戸際です。CP/Mを使う場合は，CP/Mに付属したマニュアルだけでは，決定的に不足します。また，ついでにCP/Mの勉強法について，私自身のかってな意見を述べておきます。

それは，第1に「使う」ということです。アセンブラでゲームを書きたい，とか，FORTRANを使って，数値解析をしたい，などの具体的な目標をもつことだと思います。ただ単に「CP/Mをマスターしたい」と思うだけでは，「それをマスターしても，どうってことないや」と同じことになってしまいます。以上老婆心ながら一言。

さて，本題に入ります。

CP/Mは8080，8085，Z80用のディスクオペレーティングシステムです。平たくいえば，ディスク用の便利なサブルーチン集です（少し誤解をまねくかな？）。CP/Mは，あくまでサブルーチン集ですので，それだけでは，なんにもできません。そこで，CP/Mを買うと，ASM（アスム），ED（エド），DDT（ディディティ）などの「CP/Mというサブルーチン集の上で動くプログラム」が何本か付いてきます。

ASMは，アセンブラ，EDはラインエディタ，DDTは，デバッガです。ただし，ASMとDDTは8080用ですので，Z80のニーモニックに慣れた人は，ノイローゼにかかってしまいます。

また，ラインエディタのEDは，たしかにラインエディタとしては強力なのですが，WORDMASTERのような，スクリーンエディタと比べれば，使いごこちは雲泥の差です。そこで，アセンブラを中心にしたCP/Mの使い方は，次のようになります。

❶スクリーンエディタを手に入れる

X1のCP/Mには，WORDMASTERが付いてくるので問題はありませんが，MZの

＜S（ちょっと休息）

なぜCP/Mに付いているASMとDDTは、8080用だと思いますか？　たしかに8080のほうが先にできたのですが，Z80のほうが命令は強力だし、ずっと多く出まわっているからZ80用に書き直したものを付属すべきじゃないか！　と思う人が大部分だと思います。実は裏の事情をいうと、CP/Mの産地アメリカでは多少事情が違うのです。まず、数からいくと、「8080のアセンブラは知っているが、Z80は知らない人」のほうが多いらしいのです。そして、実数はよく知りませんが、8080、8085（8080より、ほんのちょっと命令数の多いCPU）を使ったコンピュータも健在です。日本では、TK－80に使われたぐらいなのを見れば、大きな差です。さて、ここでもし日本でZ80の上位コンパチブルで、アセンブラの体系がまったく違うCPUが開発されたとしたら、どうなるでしょうか？　みんなこぞってZ80と、そのアセンブラを使うのをやめて新しいCPUのほうへ移っていくでしょうか？　きっとそうはならないでしょう。少なくともアセンブラの体系が同じなら別ですが。というわけでCP/Mの開発元のデジタル・リサーチ社もZ80用のツールを標準でCP/Mに乗せる気はないようです。ましてや、このあと述べるように、Z80用のツールが数多く市場に出まわっていますし、ソフト戦線は16ビット上で展開されているようなので。

＜S（休息終わり）

ユーザーは，買わなければなりません。WORDMASTERの場合は，38,000円です。お金のない人は，EDに慣れるしかありません(EDは慣れれば使いごこちは悪くないようです)。

❷アセンブラを手に入れる

大部分の人がZ80のニーモニックに慣れていると思いますので，やはりASMではもの足りないと思います。Z80のニーモニックが使えるとしたら，やはりMACRO-80が最良なのですが，60,000円と高価です。

❸デバッガを手に入れる

アセンブラと同じようにDDTでは，Z80特有の命令は使用できません。Z80用のデバッガとしては，ZSIDが一般的ですが，38,000円します。

以上のエディタ，アセンブラ，デバッガの3種のソフトを揃えると非常に便利なのですが，3種のソフトとも機能は低いが，一応CP/Mに付属しているということが業腹ですね。ひょっとしたら，私のことをソフト屋の回しものだと思った方もいるでしょうが，無理もないでしょう。そこで，今回は間に合いませんでしたが，次回には，この3種のソフトのうち安価そうなものを探して「お買い得」リストを書きたいと思います。もっとも，それぞれは10,000円を下ることはありませんが，それでも購入のときの参考になるでしょう。

さて，手に入れたあとのことですが，

❹まず，エディタの使い方を覚える

たとえば，WORDMASTERの命令は，表1のように数多くありますが，最初に全部を覚える必要はありません。最低限度の使用法を覚えて，その他は必要なつど覚えていけばよいのです。スクリーンエディタの場合，最低限度必要なのは次ぐらいでしょう。WORDMASTERを例にあげておきます。

a. 起動のしかた。

WM␣TEST. ASM

TEST. ASMというファイルが，存

表1　WORDMASTERのまとめ

制御コード	機　　能	制御コード	機　　能
∧O	インサートをオン/オフにします。	∧G	カーソルの位置にある文字を削除します。
∧Sまたは←	カーソルを1文字左へ動かします。	DEL	カーソルの左の文字を削除します。
∧Dまたは→	カーソルを1文字右へ動かします。	∧L	1ワードの左側を削除します。
∧A	カーソルを1ワード左へ動かします。	∧T	1ワードの右側を削除します。
∧F	カーソルを1ワード右へ動かします。	∧U	カーソルから行の左側を削除します。
∧Q	カーソルを次のタブストップへ動かします。	∧K	カーソルから行の右側を削除します。
∧Xまたは↓	カーソルを1行下へ動かします。	∧Y	復帰を含めた行全体を削除します。
∧Eまたは↑	カーソルを1行上へ動かします。	∧I	タブ。ファイル文字として扱われます。
∧V	カーソルを画面の一番上へ，次に一番下に動かします。一番下に動かすには2回タイプします。	∧N	復帰を挿入し，カーソルをその前にセットします。
		<CR>	復帰。
∧B	カーソルを行の左端へ，次に右端に動かします。右端に動かすには2回タイプします。		インサートオンのとき：復帰改行をファイルに挿入します。
			インサートオフのとき：カーソルを次の行の先頭へ動かします。
∧W	画面上でファイルを1行下へ動かします。		ファイルの終わりで：ファイルに追加します。
∧Z	画面上でファイルを1行上へ動かします。	∧@	次のコマンド文字を4回実行します。
∧R	ファイルを1画面分戻します。	∧P	次の文字（制御文字でも）をファイルに入れます。
∧C	ファイルを1画面分次へ動かします。	escape	ビデオモードからコマンドモードに移ります。

コマンドモードでVと復帰をタイプすると，ビデオモードに戻ります。

WORDMASTERコマンドのまとめ

+ー……+またはー。省略した場合は+。
$……escape，∧Z，または復帰。
n……数値。省略した場合は1。#は65535。
+ーnC……ポインタをn文字動かします。
+ーnD……n文字削除します。
+ーnL……ポインタをn行動かします。
+ーnK……n行削除します。
+ーnT……n行表示します。
+ーnP……nページ(1ページ23行)ポインタを動かし，表示します。
+ーn[CR]……ポインタをn行動かし，1行表示します。
nIテキスト$……テキストをn回挿入します。
nAテキスト$(Append)……1Lを実行するつど，テキストを挿入します。
+ーnFキー$(Find)……キーで，短い検索をします。
+ーnNキー$(Next)……キーで，長い検索をします。
+ーnSキー$新しいテキスト$……キーで短い検索をし，新しいテキストで置き換えます。
+ーnRキー$新しいテキスト$(Replace)……キーで長い検索をし，新しいテキストで置き換えます。
/F, /N, /S, /R……それぞれ，F.N.S.Rと同様です。ただし，検索が失敗したときは，現在のループあるいはQXを出て，実行を続けます。
n〈…〉……ループn回…を繰り返します。nを省略したときは，nを64Kとみなします。
Y[d:]名前・タイプ$(Yank)……ファイルを読み込みます。
nW[d:]名前・タイプ$……n行ファイルに書き込みます。
nQP……ファイルからn行削除し，それらをスクラッチパッドに入れます。
n/QP……ファイルからn行削除し，それらをスクラッチパッドに追加します。
nQG(Get)……スクラッチパッドの内容をn回ファイルにコピーします。
QT……スクラッチパッドの内容を表示します。
QK(Kill)……スクラッチパッドをクリアーします。
QX……スクラッチパッドの内容を実行します。
QLテキスト$(Loab)……テキストをスクラッチパッドに入れます。
n/QLテキスト$……テキストをスクラッチパッドに追加します。
nZ(Sleep)……n秒待ちます。
V……ビデオモードになります（ESCを押すと元に戻ります)。
n!……文字コードnをファイルに入れます。
;……これに続く文字を注釈とみなします。
E……ファイルをセーブし，編集を終了します。
H……ファイルをセーブし，編集を継続します。
Q(Quit)……それまでの編集を無効にし，編集を終了します。
O……それまでの編集を無効にし，編集を継続します。
"テキスト"の中の特殊文字：∧N　復帰改行，∧Y　escape.
"キー"の中の特殊文字：∧S　セパレータ，∧A'　任意の文字，∧Ox　x以外の文字，∧N　復帰改行，∧Y　escape。

在しない場合は，WORDMASTERが，自動的に作ってくれます。存在するのならば，そのファイルを読み込んでくれます。

b. カーソルを上下左右に動かす。
X1用のWORDMASTERは，[↑][↓][←][→]のキーを使って，カーソルを動かすことができるので，この点は簡単でしょう。∧E，∧X，∧S，∧Dも同様な機能がありますから，慣れたら，好きなほうを使うようにすればよいでしょう。

c. 挿入と削除の方法。
WORDMASTERでは，∧Oで挿入モードになります。もう1度∧Oでそのモードから抜け出します。
削除は，[DEL]だけでよいでしょう。1行挿入の場合は，挿入したい行の左端へカーソルを持っていって，∧Nです。1行削除は，削除したい行へカーソルを持っていって，∧Yです。

d. エディットしたファイルを没にして，CP/Mに戻る。
[ESC]でコマンドモードに入ります。その後Q [CR]で，Abort (Y/N) ?と聞いてきますのでY [CR]としてください。

e. エディットしたファイルを，ディスクに書き込んでから，システムに戻る。
[ESC]でコマンドモードに入り，その後E [CR]です。

さて，EDの場合は少々面倒です。

ED␣TEXT. ASMでEDを起動する。次に，すでに存在するファイルを変更する場合は＃Aで，読み込みます。ただしその際，メモリ内に読み込みきれないほど大きなファイルであれば，途中までだけです（この場合は，Wコマンドで，最初の何行かを一時的なファイルへ書き込んでバッファに余裕をもたせてから，読みきれなかった分を読み込み…という手順でエディットが行なわれる。つまり，ユーザーがメモリマネージメントをしなければいけない。EDの最大の欠点)。ファイルが十分小さいとして，話を進めましょう。最低限度覚える必要のあるのは次の7つのコマンドぐらいです。

a. Bコマンド－バッファの先頭へキャラクタポインタ (以後CP) を移動します。

b. Iコマンド－CPの位置から挿入を始めます。∧Zで挿入モードを抜け出します。

c. n:－CPを，第n行の先頭へもっていきます。

d. Tコマンド－±nTの形式で使う。
CPの位置から，±n行分を表示する。
たとえば，CPが第10行にあるならば，
－5Tで5～9行が表示される。
5Tで10～14行が表示される。
CPは移動しない点に注意！

e. Kコマンド－CPから±n行を削除する。±nは，Tコマンドと同様。

f. Qコマンド－ファイルを没にして，システムに戻る。Q－(Y/N) ?と聞いてくるのでYと答える。

g. Eコマンド－ファイルを書き込んで，システムに戻る。

以上ですが，ラインエディタというものは，ほんとうに非人間的ですね。実は私はほとんどEDを使ったことがないのですが，

表2　ニーモニック対応表　（栗山他，ASCII SYSTEMSOFT PC-Techknow 8800 Vol. 1，㈱／アスキー，pp.324-325，から）

>……Intel 8080 Mnemonics to Z-80 Mnemonics……<
r : Register (A, B, C, D, E, H, L)　n : Constant (8bits)　nn : Constant (16bits)

[i8080]	[Z80]	[i8080]	[Z80]	[i8080]	[Z80]	[i8080]	[Z80]
ACI n	ADC A, n	DCX B	DEC BC	LXI SP, nn	LD SP, nn	RPO	RET PO
ADC M	ADC A, (HL)	DCX D	DEC DE	MOV M, r	LD (HL), r	RRC	RRCA
ADC r	ADC A, r	DCX H	DEC HL	MOV r, M	LD r, (HL)	RST 0	RST 0H
ADD M	ADD A, (HL)	DCX SP	DEC SP	MOV r, r'	LD r, r'	RST 1	RST 8H
ADD r	ADD A, r	DI	DI	MVI M, n	LD (HL), n	RST 2	RST 10H
ADI n	ADD A, n	EI	EI	MVI r, n	LD r, n	RST 3	RST 18H
ANA M	AND (HL)	HLT	HALT	NOP	NOP	RST 4	RST 20H
ANA r	AND r	IN n	IN A, (n)	ORA M	OR (HL)	RST 5	RST 28H
ANI n	AND n	INR M	INC (HL)	ORA r	OR r	RST 6	RST 30H
CALL nn	CALL nn	INR r	INC r	ORI n	OR n	RST 7	RST 38H
CC nn	CALL C, nn	INX B	INC BC	OUT n	OUT (n), A	RZ	RET Z
CM nn	CALL M, nn	INX D	INC DE	PCHL	JP (HL)	SBB M	SBC A, (HL)
CMA	CPL	INX H	INC HL	POP B	POP BC	SBB r	SBC A, r
CMC	CCF	INX SP	INC SP	POP D	POP DE	SBI n	SBC A, n
CMP M	CP (HL)	JC nn	JP C, nn	POP H	POP HL	SHLD nn	LD (nn), HL
CMP r	CP r	JM nn	JP M, nn	POP PSW	POP AF	SPHL	LD SP, HL
CNC nn	CALL NC, nn	JMP nn	JP nn	PUSH B	PUSH BC	STA nn	LD (nn), A
CNZ nn	CALL NZ, nn	JNC nn	JP NC, nn	PUSH D	PUSH DE	STAX B	LD (BC), A
CP nn	CALL P, nn	JNZ nn	JP NZ, nn	PUSH H	PUSH HL	STAX D	LD (DE), A
CPE nn	CALL PE, nn	JP nn	JP P, nn	PUSH PSW	PUSH AF	STC	SCF
CPI n	CP n	JPE nn	JP PE, nn	RAL	RLA	SUB M	SUB (HL)
CPO nn	CALL PO, nn	JPO nn	JP PO, nn	RAR	RRA	SUB r	SUB r
CZ nn	CALL Z, nn	JZ nn	JP Z, nn	RC	RET C	SUI n	SUB n
DAA	DAA	LDA nn	LD A, (nn)	RET	RET	XCHG	EX DE, HL
DAD B	ADD HL, BC	LDAX B	LD A, (BC)	RLC	RLCA	XRA M	XOR (HL)
DAD D	ADD HL, DE	LDAX D	LD A, (DE)	RM	RET M	XRA r	XOR r
DAD H	ADD HL, HL	LHLD nn	LD HL, (nn)	RNC	RET NC	XRI n	XOR n
DAD SP	ADD HL, SP	LXI B, nn	LD BC, nn	RNZ	RET NZ	XTHL	EX (SP), HL
DCR M	DEC (HL)	LXI D, nn	LD DE, nn	RP	RET P		
DCR r	DEC r	LXI H, nn	LD HL, nn	RPE	RET PE		

ときには素っ裸のCP/Mシステムと対決しなければならないときがあるので，この最低限の7コマンドでかわしています。EDは，コマンドに習熟すると，かなり強力らしいのですが，慣れるにはかなりの経験が必要です。

❺アセンブラの使い方を覚える

ASMの場合は，

ASM␣TEXT [CR]（エクステンション—"・"のあとの"ASM"の3文字—は不要）で，エラーのない場合は，

TEXT. PRNとTEXT. HEXという2つのプログラムができます。～.PRNのほうはアセンブルリスト，～.HEXのほうは，機械語に半分なりかかったデータが入ってます。次に，

LOAD TEXT [CR]

で，TEXT. COMというファイルができ，それが最終的にできる実行可能形式のファイルです。アセンブラの最初にORG␣100H（Hをおとさないように！），最後にENDを忘れないように。

MACRO-80の場合は，少々面倒になります。まず，～.PRNのファイル出力のためにプログラムの先頭に.PHASE␣100Hと書いておいてください。ORG命令はなしです。このソーステキストをアセンブルするには，

M80␣TEXT, TEXT=TEXTとします。これで，TEXT. PRN, TEXT. RELの2つのファイルができます。～.RELとは，リロケータブルファイルで，LINK-80を使って，好きな番地に配置できるようになっています。次に，

L80␣/P：100, TEXT, TEXT/E/N

で，TEXT. COMができます。

実はMACRO-80は，普通はこのような使われ方はしません。それを無理に，ASMと同じように使っているので，多少面倒な手続きになりますが，決まりきった形式なのですぐに覚えるでしょう。MACRO-80は，非常に強力なアセンブラで，多分これ以上のものは必要ないはずです。

❻デバッガの使い方を覚える

このデバッガに類したものは，市販されているので，使用法を覚えるのは簡単でしょう。DDTの場合は，DDT␣TEXT. COMで，DDTが起動し，同時にTEXT. COMもメモリ内に読み込まれます。具体的な例はあげません。

ZSIDは，Z80のニーモニックが使える以外はDDTとほぼ同じなのですが，アセンブルする際にシンボルテーブルを作っておくことにより，ラベル名を参照しながら，デバッグをすることができます。この点DDTでは絶対番地しか参照できなかったのと大きな差があります。しかし，私の考えでは，特にたちの悪いバグ以外では，この機能はパワフルすぎる気もしますが。

以上が，アセンブラを中心にしたCP/Mの使い方の基本です。いちばん大事なのは，簡単な方法でもよいから，とにかく使えるようになることです。最初から，すべてのテクニックを使おうとしても不可能です。

今月は，なんだかんだ言いながら，たいしたことはできませんでした。そこで，あくまで，ASMで耐え忍ぶ人のために，8080→Z80の対応表（表2）と，8080ニーモニックで書いたサンプルプログラムを載せておきます。一度対応を覚えれば，8080でもけっこうやれるものですので，気を確かにもってください。サンプルプログラムのほうは，ソースだけですので，頑張って打ち込んでください。きっとエディタの練習になるでしょう。また，打ち込みミスでバグも入るでしょうから，デバッガの練習になるでしょう。ああ，なんたる親心！

それでは^C（さようなら）

今月の問題

サンプルプログラムを好きなように書き換えて，プログラムが起動したのちにも，ドライブを選択できるようにせよ。その際，プログラムの起動前と，^Cで抜け出したあとで，ON-LINEのドライブを変えないようにするためにはどうすればよいか。

サンプルプログラムの説明

CP/Mでは、1セクタは128バイトです。これはロジカルセクタと呼ばれます。X1，MZのCP/Mでは、本当のセクタ（フィジカルセクタ）は256バイトなので、ソフトウエアで、2つに分割していることになります。今月のプログラムは、直接にBIOSをCALLして、ロジカルセクタを読み出し、ダンプさせます。100行目からのCALLERというサブルーチンでは、少々異常なことをしていますので、気をつけてみてください。ATOIは、文字列を数値にして、HLに入れてリターンします。その際キャリーフラッグが立っていたら、おかしな文字列だったということです。PRINTAは、Aレジスタの中身を2桁の16進数で表示します。PCHARは、Aレジスタの中身をアスキーコードとする文字を表示します。CRLFは行がえをします。それぞれ汎用性のあるサブルーチンです。

使用法は起動すると、トラックとセクタの番号を聞いてきますので、それぞれ入力してください。このとき^Cを入力するとシステムに戻ります。大きすぎる数値を入れるとエラーが出ますよ。

また、ダンプするドライブはON-LINEのドライブに限定されてますからB>A：DDUMPのようにして、Bドライブをダンプしてください。

サンプルプログラム

```
1: ;        Disk Dump Program for all machine
2: ;
3: ;        Copyright (C) 1984 March  Youichi Takano
4: ;
5:          ORG     100H
6: WBOOT    EQU     0000H   ;return to ccp
7: BDOS     EQU     0005H   ;bdos entry addr.
8: CONS     EQU     1       ;read char
9: TYPEF    EQU     2       ;print 1 char
10: PRINTB  EQU     9       ;type buffer
11: READB   EQU     10      ;read buffer
12: DSKBUF  EQU     080H    ;disk buffer addr.
```

```
13: ;
14: CR      EQU     0DH
15: LF      EQU     0AH     ;line feed
16: ;
17: ;       SP initialize "
18:         LXI     SP,MYSP ;user SP
19: ;
20: MLOOP:  CALL    ASKTRK
21:         MOV     A,L
22:         STA     TRKN    ;copy truck number
23: ;
24:         CALL    ASKSCT
25:         MOV     A,L
26:         STA     SCTN    ;copy sector number
27: ;
28:         CALL    DUMPGO
29:         JMP     MLOOP
30: ;
31: ;       exit only when ask TRK,SCT
32: ;
33: ;       end of main loop
34: ;
35: ;       get truck number
36: ASKTRK: LXI     D,MESTRK        ;ask truck No.
37:         CALL    INPN    ;get string
38:         JC      ASKTRK  ;if strange then again
39:         RET             ;HL = number
40: ;
41: ASKSCT: LXI     D,MESSCT        ;ask sector No.
42:         CALL    INPN    ;get string
43:         JC      ASKSCT  ;if strange then again
44:         RET             ;HL = number
45: ;
46: DUMPGO: LDA     TRKN    ;get truck No.
47:         MOV     C,A
48:         LXI     H,10
49:         CALL    CALLER  ;set truck
50: ;
51:         LDA     SCTN    ;get sector No.
52:         MOV     C,A
53:         LXI     H,11
54:         CALL    CALLER  ;set sector
55: ;
56: ;       DMA is already set (0080H)
57: ;       LXI     B,DSKBUF
58: ;       LXI     H,12
59: ;       CALL    CALLER  ;set DMA addr.
60: ;
61:         LXI     H,13
62:         CALL    CALLER  ;read sector
63:         ORA     A
64:         JNZ     ERROR
65: ;
66:         CALL    DUMPBF  ;dump 128 bytes
67:         RET
68: ;
69: DUMPBF: CALL    CRLF    ;next line
70:         LXI     H,DSKBUF        ;buffer addr.
71:         MVI     C,08H   ;y counter
72: ;
73: DLOOPY: PUSH    H       ;save HL for ascii dump
74:         MVI     B,010H  ;x counter
75: DLOOPX: MOV     A,M     ;A=(HL)
76:         INX     H       ;HL=HL+1
77:         CALL    PRINTA  ;print 1 byte
78:         MVI     A,' '
79:         CALL    PCHAR   ;1 space
80:         DCR     B       ;B=B-1
81:         JNZ     DLOOPX
82: ;
83:         MVI     B,10H   ;counter
84:         POP     H       ;get HL for ascii dump
85: ADUMP:  MOV     A,M
86:         INX     H
87:         CPI     ' '     ;printable?
88:         JNC     DUMP1   ;bigger than space then print
89:         MVI     A,'.'   ;else print '.'
90: DUMP1:  CALL    PCHAR   ;print ascii
91:         DCR     B
92:         JNZ     ADUMP
93:         CALL    CRLF
94: ;
95:         DCR     C
96:         JNZ     DLOOPY
```

```
 97:          RET
 98: ;
 99: ;        direct BIOS call HL=order of jump table
100: CALLER:  DCX     H
101:          PUSH    D
102:          MOV     D,H
103:          MOV     E,L
104:          DAD     H
105:          DAD     D
106:          XCHG
107:          LHLD    WBOOT+1
108:          DAD     D         ;now HL has addr. to jump
109:          POP D
110: ;
111:          PCHL    ;JP (HL)
112: ;
113: ;        print out (DE) and set number in HL
114: INPN:    MVI     C,PRINTB          ;print string function No.
115:          CALL    BDOS
116:          XRA     A
117:          STA     BUFF+2  ;clear top of string
118:          LXI     D,BUFF  ;DE has addr of string buffer
119: ;
120:          MVI     C,READB ;input string func. No.
121:          CALL    BDOS
122:          MVI     A,LF
123:          CALL    PCHAR
124: ;
125:          LXI     H,BUFF+1          ;load addr. of count and string
126:          CALL    ATOI              ;conv it to number
127:          RET
128: ;
129: ;        if number is strange then return with CARRY FLAG
130: ATOI:    MOV     A,M               ;get count
131:          ORA     A                 ;check count
132:          JZ      BATOI             ;bad number,return
133:          MOV     B,A               ;b=count
134:          INX     H                 ;hl points string
135:          LXI     D,0               ;clear DE
136: ALOOP:   MOV     A,M               ;get char
137:          INX     H                 ;HL points next char
138:          SUI     '0'
139:          JC      BATOI             ;too small
140:          CPI     10
141:          JNC     BATOI             ;too big
142:          PUSH    H                 ;save HL
143:          MOV     H,D
144:          MOV     L,E               ;copy DE to HL
145:          DAD     H                 ;*2
146:          DAD     H                 ;*4
147:          DAD     D                 ;*5
148:          DAD     H                 ;*10
149:          MVI     D,0
150:          MOV     E,A               ;get 1 number
151:          DAD     D
152:          XCHG                      ;DE=DE*10+BC
153:          POP     H                 ;back string address
154:          DCR     B                 ;dec count
155:          JNZ     ALOOP             ;NZ then loop again
156: ;
157:          XCHG                      ;return val is HL
158:          ORA     A                 ;any way carry=0
159:          RET
160: ;
161: ;        set carry and return
162: BATOI:   STC
163:          RET
164: ;
165: ;        print out A
166: PRINTA:  PUSH    PSW     ;save AF
167:          RAR
168:          RAR
169:          RAR
170:          RAR
171:          CALL    PRNA1
172:          POP     PSW     ;get AF
173:          CALL    PRNA1
174:          RET
175: ;
176: PRNA1:   ANI     0FH
177:          ADI     '0'
178:          CPI     '9'+1
179:          JC      PRNA2   ;ok print
180:          ADI     7       ;case A-F
```

```
181: PRNA2:  CALL     PCHAR
182:         RET
183: ;
184: PCHAR:  PUSH B ! PUSH D ! PUSH H        ;save them
185:         MVI      C,TYPEF
186:         MOV      E,A
187:         CALL     BDOS
188:         POP H ! POP D ! POP B    ;get them
189:         RET
190: ;
191: ;       next line
192: CRLF:   MVI      A,CR
193:         CALL     PCHAR
194:         MVI      A,LF
195:         CALL     PCHAR
196:         RET
197: ;
198: ;
199: ERROR:  MVI      C,PRINTB        ;print buffer function
200:         LXI      D,ERRMES
201:         CALL     BDOS
202:         RET
203: ;
204: ;
205: ;       fixed message area
206: MESTRK: DB       'Input truck number $'
207: MESSCT: DB       'Input sector number $'
208: ERRMES: DB       'DISK READ ERROR!',CR,LF,'$'
209: ;       variable area
210: TRKN:   DS       1          ;save truck No.
211: SCTN:   DS       1          ;save sector No.
212: ;
213: ;       input buffer area
214: BUFF:   DB       5,0                ;max 5 char
215:         DS       10                 ;enough
216: ;
217: ;       stack area
218:         DS       64                 ;reserve 32 level stack
219: MYSP:
220: ;
221:         END
```


通産省国家資格。情報処理技術者。

国家試験 対策通信講座

●早い準備で合格へ！今からスタート。あなたも！

★受験予定の方は必読！合格資料を無料呈！

合格資料無料呈！
●ハガキか電話で直接
下記までどうぞ！

●情報処理技術者は不足！
コンピュータの技術進歩と設置台数の急増により、情報処理技術者は不足！

●国家試験受験準備講座を開講中！
実戦即応の教材と実戦模試・ポイント指導で合格力を養成。早い準備で合格へ。

●将来性のある情報処理技術者！
企業・官庁で有資格者優遇。学生に最適。

●受験資格は不問。誰でも可！
受験資格については、学歴・年齢・性別・経験などは問いません。試験地は全国9都府市で行われます。4科目の筆記試験。

●第2種情報処理技術者国家試験の合格のポイントは、早い準備と限られた時間をムダなく効率的に活用することです。

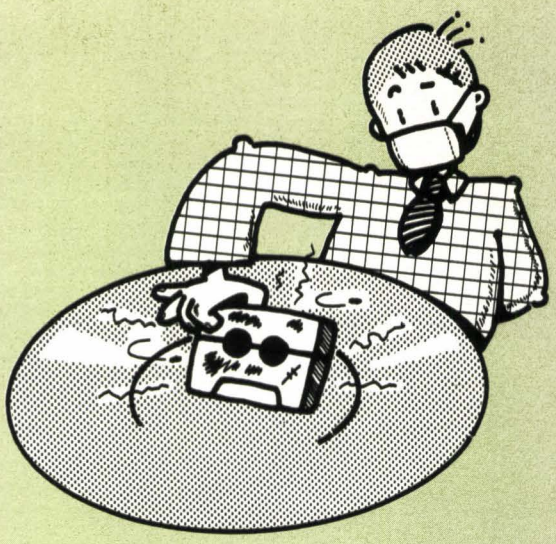
●資料希望の方は、ハガキか電話で！

(財)実務教育研究所 受験講座情報係
160 東京都新宿区大京町4の383
☎東京 03(357)8153

皿までどーぞ

清く正しく面白く

Ippei Iwai
祝　一平



はっきり言うと，日本のソフトウェアは，過去の，作ればなんでも売れるのだ，だから作るのだのがっぽがっぽ的状況から脱しつつあるようである。これはとても良いことなのだが，そこはそれ，スネにキズ持つこの業界の前科前歴からしても，あいかわらず「その筋」のソフトがでかい顔で，まだそこらあたりを転がりまわっているのである。一概にその筋と呼んでも，さまざまな筋があり，うかつに語ることはできないのである。だから，あんまりうかつにじゃなく語ることにするのである。

さて，私がソフトを見るときの基準は，今月の副題のように，清く正しく面白くである。読者の中には「面白くに関してはゲームのことじゃねーか。ビジネス関係はどうするんでい」とクロスカウンターをおくってくる諸兄がいるかもしれない。しかし，私はビジネスソフトにも別質の面白さがあるのだ，と言ってダブルクロスカウンターで返り討ちにするのである。この場合読者から，トリプルクロスという恐怖の必殺技が出現する可能性もあるが，それはおいておく。一般に，良いソフトはやはり面白い。

そういうわけで，本題に入るのである。

第1章　清いソフトとは，いかなるものなのか

清いというのは，姿勢の問題なのだ。つまり，技術以前の問題である。

さて，過去の話ではあるが，昔は清いソフトは少数であった。実名をあげるのは非常にその筋なのでさし控えるが，たとえばBASICで書かれたゲームで，走らせると平気でエラーを出すソフトが過去にはあった。その他にもリストを見ると，やたらとREM文が入っていて，しかもそれらがほとんどプログラムと関係がない，というものもあった。そのREM文たるや，某タレントの名前を挙げて「スキ！」とか書いているのである。これなどは，まさしく上げ底ソフトと呼ぶべきものであった。そのプログラムはテープで供給されていたから，ロード時間によってプログラムサイズがわかるのである。そして，あまり早くロードが終わると，それだけでその筋と判明してしまうのだ。そこで，陰湿REM文作戦が敢行されたのではないだろうか？

いやいや，状況はもっと緊迫縄縛りかもしれない。

たとえば，そのプログラムを作ったのがアルバイトだとしたら，どうなるだろうか。そして，ソフトハウスのその筋の経営者が「豚バラ肉薄切り130円」と同様に，「ゲームプログラム1バイト2円20銭」で買い上げていたとしたら，どうだろうか？　ひょっとしたら，そのような悪夢のごとき構図があったのかもしれないのである。

その他にもいろいろな清くないソフトが存在していた。たとえば，あるものは，デモのタイトル画面はやたらと豪華絢爛で，しかし肝心のゲームのほうは，その筋だったりした。ゲーム名にも同様なことがあり，期待させるネーミングで，実は中身はほとんど困りものというものもあった。広告にも同じことがあった。ビジネスソフトの場合は，「最高級」とか「万能」とかの言葉を連ねて，売ってしまえばこっちのものとかいうソフトである。私はこれらのことが，本当に過去のものになってほしいと思う。

最近増加しつつあるのが，「マシン語」と銘打った，BASICコンパイラで書かれたゲームである。たしかにそれはマシン語には違いないが，なぜ正直にコンパイラのオブジェクトだといわずに，マシン語と表示するのか？　これは清くない代表例だと思うのだ。業界各位に方針改善を願いたい。

これらの人間不信養成ソフトは，タダでも欲しくないと申しあげたい。ひとつその点をよろしくお願いしたいものだ。

第2章　正しいソフトの傾向と対策

正しくないソフトの代表例は，不親切設計である。まあ，しかたがないといって許せるが，これが一番多くもある。たとえば，やたらと騒々しいゲームである。特に困るのが，殺られたときに葬送行進曲などをシツコク演奏するやつである。パックマンの筋で，「キュ〜〜」ぐらいでよいのである。この点で悲惨なのがPC系の機械である。一部の機種以外は，ボリュームがないので，ねちねちとやられると，もはやガード下の世界が広がるのであった。それもブザーをON/OFFして出す，その筋のギャリギャリ音だから，夜中だと効果音ではなく，逆効果音になってしまうのだ。異教PC派の方々の中には，改造でしのぐ人もいると聞く。一部には音を出力させるかどうか選択させるものもあり，これはたいへん好ましいことである。また，システムプログラムの場合は，エラー時のビープ音がしつこいというもの

もある。あれは短く「ピッ」と鳴るだけでよいのだ。それから，PSGを使っている場合は，音が出っぱなしになることもある。デモで，「パンパカパーン」と鳴っている途中で，スタートさせると，「パ――――」とゲームの間中鳴りつづけるというものである。

音だけならば，まだまだ許容範囲にあるが，これがソフトの本質，たとえば操作性などにかかわってくると，イライラの原因になってくる。たとえば「遅い」プログラムである。

一口に遅いといっても，いろいろな遅さがある。思いつくままに挙げると，①キーの反応が遅い。②特定のルーチン――たとえば変数のイニシャライズが遅い。③周辺器機へのアクセスが遅い。④画面処理が遅い。⑤とにかく全部遅い，などがある。時々特別に複雑なことをしているわけでもないのに異常に遅いプログラムがある。こんなもん，よく売っていやがる，というレベルもある。遅すぎたら発表しないぐらいの気持ちを持ってほしいと思うのである。とくにビジネスソフトなどでは，値段が万以上なのに，遅かったりするソフトに出会うと，イラダチは風雲を呼び，怒濤となって砕け，私は思わず，「ぎー！」と叫びつつキーボードを殴ったりするのである。ただし，これは編集室にあるサンプルを試用中の話だから，万一私が自腹を切って買ったものがその筋だったら，決して逆上などはせずに，ひたすらに凶眼を光らせながらソフトハウスへの物理的報復を検討したりする可能性もあるので，ソフトハウスの方々は気をつけて作ってもらいたい。

以上パソコン騒音と，テレンコソフトは，諸悪の2大潮流である。その他小悪を挙げるなら，パックマンのような迷路型ゲームで，その通路での右左折が異常に難しいものがある。つまり，ピタッと位置を合わせてから，左右に対応するキーを押さないと曲れないという状況である。少しでもずれていると，もうだめなのである。最近では枝をはった木を迷路にみたてたやつの唯一の欠点がこれである。実に残念である。

それから，プログラムを単純にするために，おかしな制限をしているものもある。入力出のディスクドライブを固定したり，文字列の入力で DEL キーが使えなかったり，数値入力をテンキーからできなかったりである。

これらのことは，小さな手抜きであるから，正しくない，とまで言いきれない。しかし，セコさは元から断つべきなのだ。どうかひとつ，愛と感動のソフトをめざしていただきたい。

第3章 面白くなければソフトではない

たとえば，アドベンチャーゲームの場合である。このジャンルで発表されているものは，だいたいにおいて自由度の低い一本道構造になっているようである。私はその点については異存がない。しかしその筋のアドベンチャーゲームの場合は，そのカギが非合理的なほど難解だったりする。ある例では，固有名詞が要求されるのだが，まったくのノーヒントで，推理は不可能に近かった。たまたま記事用にソフトハウスより解答がきていたのでわかったのだが，一般の方々はどうしたんだろうか？　簡単なアドベンチャーゲームほどくだらないものはないが，かといって難しさにも質と限度が必要なのである。偶然の発想でしか解けないようなものは，解けても全然楽しくないのである。

さて，英語でコマンドを入力するタイプのアドベンチャーゲームにも問題がある。第一が語句解析の貧弱さである。同義語を受けつけないぐらいは序の口で，絶対に英語じゃない！　という文章を要求する場合もあるのだ。

次に反射神経型ゲームのほうだが，その筋のゲームは，次の7種類ぐらいに代表されるのである。

❶ 自動車教習所型ゲーム。技術ばっかりで，なんの面白みもないゲームである。たとえば，赤いエイリアンが出てきたら，右端に行って一定の距離まで引きつけてから射たないと，必ずやられる，とかのたぐいである。こんなのをやるよりも，さっさと寝てしまったほうがよい。

❷ ルーレット型ゲーム。単に確率の問題だけで，ほとんど腕の見せどころのないゲームである。アルゴリズムを乱数で代用するとできる。

❸ ガンダム型ゲーム。まったく操作法を理解できないゲームである。やたらと多くのキーを押さねばならなかったりする。ニュータイプでなければ楽しめない。

❹ 竹槍特攻型ゲーム。ある場面までくると突然敵が無敵になり，どうしてもクリアできなくなる。馬鹿にされているようで，作者に殺意を抱いてしまう。

❺ 哀愁無限地獄型ゲーム。まったく同じパターンの速度が上るだけ，もしくは敵の数が変わるだけで，エンエンと繰り返される。これをやっていると，人生の意味について，考え始めたりする。

❻ シュールレアリズム型ゲーム。まったく常軌を逸した状況設定のもとで行なわれるゲーム。必ずしも面白くないとはいえないが，わざとらしくてバカバカしくなる。

❼ IQ検査型ゲーム。時間制限のある思考型ゲーム。ただし時間は十分にはほど遠い。これをやると，暴力中学生の気持ちがしみじみと理解できる。

だいたい以上である。

ビジネスソフトに関していえば，平凡なんの哲学もないソフトが一番うっとうしいのである。最低限度の機能を，繁雑な命令で実行しなければならないものや，一見強力そうだが，実はまったく役に立たないDOSなどは言語道断である。高価なくせになにをやっとるのだと言ってしまうのである。

第4章 だから，どーだというのだ

2，3年前だけど，たしかグラフィックを使ったソフトが出回り始めたころ，それでもゲームはスタートレックが最高だった。なぜ，文字だけのゲームがあんなに私達を興奮させたのだろうか？　それはきっと，良いソフトとはかわいいキャラクタなどではなく，もっと別のものであるということだと思う。そして最近は機械語のソフトが増えているが，間もなく速いだけではなんにも面白くないという時代がやってくるのだと思う。ユーザーがはてしなく舌を肥やし，そのユーザーを満足させる形でソフトが発展することを望むしだいなのだ。

X1面白テクニック

VTR編集用ユーティリティ

Hiroto Ashikaga
足利裕人

はじめに

前回は，スーパーインポーズを使用してのVTR編集にもっとも基本となるタイトルや，字幕の作成をしました。今回は，VTR編集をより楽しく充実したものとする，画面装飾用スーパーインポーズ・ユーティリティを紹介します。

X1のグラフィック機能を生かし，作画用プログラムを用いて小劇場風の演出を行なったのが写真1です。これにタイトルを表示し，幕を開閉させたりする機能をつければより豪華になります。子供向けテレビ紙芝居や，社内カラオケ大会などに使うのも楽しいでしょう。ここで紹介するのは，画面に矢印，円，ハートのマークを表示するプログラム“MARK”と，画面に鍵穴，ハートの窓，多角形の窓を開け，そこからVTR画面をのぞく“HOLE”，場面の切り換え効果を高める“SHUTTER”，そして野球を本格的に演出する“SCORE WRITER FOR BASEBALL”です。前回のプログラムと併せると，基本的VTR編集がほとんどカバーできるようになります。

■❶ MARK

リスト1のプログラムをRUNすると，画面に矢印・円・ハートのマークの選択が表示されます。矢印のマークは主にCAI用に使うもので，写真2はラグビーの試合中のボールの動きを追ったものです。試合の作戦を立てることにも使えますし，また教育番組を録画して，授業中にX1を操作しながら重要な物体の動きを示すことができますね。このプログラムでは，テンキーを用いて，各マークを上下左右斜め方向に移動させることができます。リスト1の40行のB$が矢印のDATAを与えます。

円のマークは矢印と同じ目的で使われます。写真3はテレビ画面に円をスーパーインポーズしたものです。円のマークの選択後，円の直径をcm単位で入力すると，その値に応じた大きさの円が画面に現われます。テンキーで必要な位置へ移動させます。

ハートのマークは，結婚式や新婚旅行の録画ビデオなどの編集用に使えますね。ハートのグラフィックデータ（リスト1の120行，130行）を蝶に変えて，家族ハイキングの録画シーンの上を飛ばしたり，マンガの主人公やキューピッドを移動させながらスーパーインポーズすると楽しい記録テープができあがることでしょう。

●ワンポイントテクニック

1ドットごとのグラフィック・パターンの移動を行なうPOSITION，PATTERN命令を使用し，動きを速くするため，上下方向に5ドットごとの移動をキー入力で行なっています。テンキーは，[5]を中心として8方向に使われています。110行のB$が矢印，120行のD$，E$がハートの左半分のグラフィックデータです。MIRROR$命令を用いて右半分を作り出し，左右対称なハートを作り出しています。皆さんもグラフィックテクニックのひとつとしてぜひ活用してください。各パターンの色は，カラーコードの7色から選びます。680行のCANVAS文で，マルチ画面モードのグラフィック1のカラーコードを決定し，700行のSCREEN文で，グラフィック1のみドットセット・リセットを行ないます。こうすることによって3個のパターンについて，どれにも共通なカラー発生用サブルーチンになります。スーパーインポーズを中断，または終了させるにはスペース・キーを押してください。パターンが消え，ビデオ画面のみ録画されます。

写真1
小劇場風にスーパーインポーズ

リスト1　MARK

```
10 '**********************************
20 '*                                *
30 '* X1 VIDEO UTILITY (MARK) V.1 *
40 '*                                *
50 '* Sep. '83 COPYRIGHT (C)       *
60 '*            by H.ASHIKAGA      *
70 '*                                *
80 '**********************************
90 '
100 WIDTH 40:INIT:PALET:SCREEN 0,0:CLS4
110 B$=HEXCHR$("080C0E0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0F0000000080C0E0F0
F8FCFEFF80808080C0C0C0E0E0E0F0F0")
120 D$=HEXCHR$("00000000000000103070F0F1F1F3F3F3F7F7F7FFFFFFFFFFFFFF7F7F3F3F1F0701
")
130 E$=HEXCHR$("01071F3F7FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFEFCF8F0E0C0
")
140 D1$=D$:E1$=E$
150 F$=MIRROR$(E1$):G$=MIRROR$(D1$)
160 C$=D$+E$+F$+G$
170 COLOR 7,0
180 LOCATE 7,2:PRINT"■■  SELECT MARK  ■■"
190 LOCATE 10,6:PRINT"ﾔｼﾞﾙｼ ... 1"
200 LOCATE 10,8:PRINT"ｴﾝ    ... 2"
210 LOCATE 10,10:PRINT"ﾊｰﾄ   ... 3"
220 COLOR 4:LOCATE 6,16:PRINT"** ﾅﾆ ｦ ﾋｮｳｼﾞ ｼﾏｽｶ "::INPUT I:CLS
230 IF I<1 OR I>3 THEN 170
240 ON I GOTO 250,310,380
250 'ﾔｼﾞﾙｼ
260 X=160:Y=100:X$=B$:P=24
270 GOSUB "ｲﾛ"
280 X=160:Y=100:X$=B$:P=24
290 POSITION X,Y:PATTERN P,X$
300 GOSUB "INP"
310 LABEL "ｴﾝ"
320 GOSUB "ｲﾛ"
330 LOCATE 4,10:INPUT"ｴﾝ ﾉ ﾁｮｯｹｲ (cm) ";RR:CLS
340 R=RR*6
350 X=160:Y=100:CIRCLE(X,Y),R
360 GOSUB "INP"
370 :
380 LABEL "ﾊｰﾄ"
390 GOSUB "ｲﾛ"
400 X=150:Y=100:X$=C$:P=32
410 POSITION X,Y:PATTERN P,X$
420 GOSUB "INP"
430 END
440 :
450 LABEL "INP"
460 A$=INKEY$
470 IF A$=" " THEN 630
480 IF A$="" OR A$<"1" OR A$>"9" THEN 460
490 IF A$="8" THEN Y=Y-5
500 IF A$="7" THEN Y=Y-5:X=X-5
510 IF A$="2" THEN Y=Y+5
520 IF A$="9" THEN Y=Y-5:X=X+5
530 IF A$="4" THEN X=X-5
540 IF A$="1" THEN X=X-5:Y=Y+5
550 IF A$="6" THEN X=X+5
560 IF A$="3" THEN X=X+5:Y=Y+5
570 IF I=1 THEN GOSUB "ALR"
580 IF I=2 THEN GOSUB "CIR"
590 IF I=3 THEN GOSUB "HER"
600 CLS4
610 IF I=1 OR I=3 THEN POSITION X,Y:PATTERN P,X$ ELSE CIRCLE(X,Y),R,C
620 GOTO 460
630 CLS4
640 GOTO 640
650 LABEL "ｲﾛ"
660 CLS
670 LOCATE 4,8:INPUT "ｲﾛ ｦ ｴﾗﾝﾃﾞ ｸﾀﾞｻｲ (1-7) ";C
680 CLS:CANVAS C,1,2
690 IF C<1 OR C>7 THEN 670
700 SCREEN 0,0,1
710 RETURN
720 LABEL "ALR"
730 IF X<0 THEN X=0
740 IF X>304 THEN X=304
750 IF Y<24 THEN Y=24
```

写真2 矢印の表示（"MARK"）

```
760 IF Y>198 THEN Y=198
770 RETURN
780 LABEL "CIR"
790 IF X<5 THEN X=5
800 IF X>315 THEN X=315
810 IF Y<5 THEN Y=5
820 IF Y>195 THEN Y=195
830 RETURN
840 LABEL "HER"
850 IF X<5  THEN X=5
860 IF X>285 THEN X=285
870 IF Y<35 THEN Y=35
880 IF Y>195 THEN Y=195
890 RETURN
```

写真3 円の表示（"MARK"）

❷ HOLE

このプログラムでは，3種類の窓を開けることができ，窓の形や周囲の色によって同じ場面でも異なった効果を与えることができます。

プログラムをRUNすると，背景色（各カラーコードに対応）と窓の大きさ（5段階）の入力を求めてきます。入力終了後画面は背景色一色になり，各形の窓がそれぞれのサブルーチンによって描かれ，その内部が青で塗りつぶされます。この青をPAINT文で黒に変えてスーパーインポーズすると，窓からビデオやTV画像がのぞけるようになります。写真4はハート形の窓からのぞいたものです。あらかじめ撮影しておいた写真を，ビデオカメラで撮影し，デジタルテロッパーでハート形のぞき窓と合成し，この画面を写真撮影すると，素敵なポートレートになります。また印画紙になったハガキを使って素晴らしいラブレターや家族の近況報告，新婚旅行から帰ってからの挨拶状などいかがでしょう。

写真5は，鍵穴ののぞき窓で，何か秘密めいた効果が表われてきます。穴の大きさを1から5まで変化させ，ビデオカメラをそれにつれて被写体にだんだん近づけていくと，より効果が上がります。

写真6は多角形ののぞき窓で，X1特有のPOLY命令を使い，三角形以上の多角形を作り出しています。その場面に応じた多角形を使いわけると楽しいでしょう。

●ワンポイントテクニック

このプログラムでは，ビデオからの映像がコンピュータ画面の端からはみ出ないように，テレビの垂直振幅COMPを調整しておいてください。鍵穴はCIRCLE文，LINE文で描きます。ハートの形は，350行の極座標を用いたオリジナルの関数を使って描きました。独自の図形によるユニークな窓を作ると楽しいですよ。バックカラーを指定したときの，窓の開け方のテクニックを参考にしてください。

リスト2　HOLE

```
10 '*************************************
20 '*                                   *
30 '* X1 VIDEO UTILITY (HOLE) V.1       *
40 '*                                   *
50 '* Sep. '83 COPYRIGHT (C)            *
60 '*            by H.ASHIKAGA          *
70 '*                                   *
80 '*************************************
90 '
100 WIDTH 40:CLS:COLOR 7,0
110 LOCATE 8,2:PRINT"**  SELECT HOLE  **"
120 LOCATE 8,6:PRINT"1 ... KEY HOLE"
130 LOCATE 8,8:PRINT"2 ... HEART HOLE"
140 LOCATE 8,10:PRINT"3 ... POLYGONAL HOLE"
150 LOCATE 8,14:COLOR 4:INPUT"HIT 1-3 KEY  ";I:COLOR 7
160 CLS:ON I GOTO 170,300,420
170 'KEYHOLE
180 INIT:PALET:CLS 4
190 GOSUB "INP"
200 CIRCLE(160,70),25+5*K,1
210 PAINT(160,70),1,1
220 LINE(160,80)-(130-5*K,140+8*K),PSET,1
230 LINE-(190+5*K,140+8*K),PSET,1
240 LINE-(160,80),PSET,1
250 PAINT(160,140),1,1
260 PALET 1,0
```

写真4ハートの穴より覗いて編集

```
270 LABEL "END"
280 IF INKEY$="" THEN 280
290 END
300 '
310 'HEART
320 INIT:PALET:CLS 4
330 GOSUB "INP"
340 FOR A=0 TO 2*π STEP π/400
350   R=(50-20*COS(A))*COS(A)+50
360    Y=R*COS(A):X=R*SIN(A)
370   PSET(160+X*(1+K/8),60+Y*(1+K/8),1)
380 NEXT A
390 PAINT(160,65),1,1:PAINT(160,120),1,1
400 PALET 1,0
410 GOTO "END"
420 'POLY
430 INIT:PALET:CLS 4
440 LOCATE 2,2:INPUT "ﾅﾝｶｯｹｲ ﾃﾞｽｶ ";N
450 IF N<3 THEN 440
460 GOSUB "INP"
470 POLY(160,100),40+10*K,1,360/N,90,(N-1)*180+90
480 PAINT(160,100),1,1
490 PALET 1,0
500 GOTO "END"
510 LABEL "INP"
520 LOCATE 2,5:INPUT"BACK COLOR ﾃﾞ ｴﾗﾝﾃﾞ ｸﾀﾞｻｲ (1-7)";BC
530 IF BC<1 OR BC>7 THEN 520
540 LOCATE 2,8:INPUT"ｵｵｷｻ ﾃﾞ ｴﾗﾝﾃﾞ ｸﾀﾞｻｲ (1-5)";K
550 IF K<1 OR K>5 THEN 540
560 COLOR ,BC:CLS
570 RETURN
```

写真5 HOLE実行例（KEY HOLEモード）

写真6 HOLE実行例（POLYGONALモード）

■❸SHUTTER

1コマごとに編集できるような編集機を持っていないと，場面の転換時の画面の乱れが避けられません。これを防ぐには，切り換え時にコンピュータ画面に中継ぎを演じてもらう必要が生じます。このプログラムは，放送局で使用している効果をX1で簡単に行なうためのものです。

RUNするとOPENとSHUTの選択を求めてきます。ある場面が終わるときにSHUTを選び，場面が始まるときにOPENを選びます。次に画面を覆い尽くす色を選びます。このとき背景色も同じ色になり，VTR画面を完全にカバーします。

次に画面は，リスト3の180行から250行にあるコメントを表示します。1から6のキーで，その場面にあったSHUTTERを選んでください。写真7はAT RANDOMのモードを選んだもので，OPENモードになっています。750行からのタイマーサブルーチンを変更して時間間隔を長くとり，VTRからの動物や花の映像を映しながらランダムに少しずつのぞき窓を開け，名前を当てさせるという子供用CAIや，家族向けクイズなどに利用してはいかがでしょうか。写真8はSCREWのモードです。中央から回転しながらOPENまたはSHUTしていきます。このSHUTTERのプログラムは，編集時の使用頻度のもっとも高いもののひとつです。

●ワンポイントテクニック

プログラムを実行すると，すぐ左上に青い四角が塗りつぶされるのが瞬間見えます。これは310行からのサブルーチンで，横25ドット×縦40ドットの領域をGET文で格納するからです。したがって，この領域をPUT文で表示することによりSHUTモードが，また青をPALET文で黒に変えることによりOPENモードが実行されます。SHUTモードでは，640×400ドットの領域をすべてカバーしたあと，はみ出た映像を同色の背景色で覆って消します。OPENモードでは逆に最後に背景色を消し，画面全体にVTRの映像を映し出すようにしています。SCREWモードで，中央かららせんを描く方法をいろいろやってみましたが，780行のように次に進む方向をDATA文で示す方法が一番簡単なようでした。図にDATAと移動方向の関係を示します。

写真7 ランダムに穴を開け，場面を切り換える。

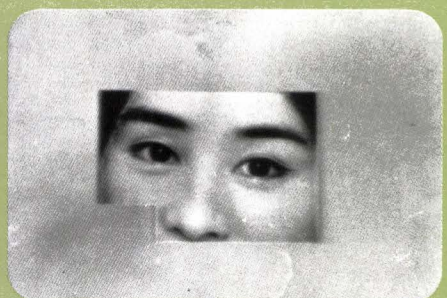
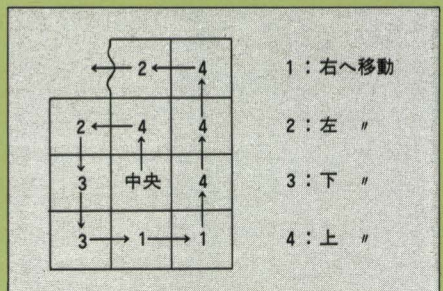


写真8 SHUTTER実行例（SCREWモード）

図　移動DATA対応表



リスト3 SHUTTER

```
10 '**********************************
20 '*                                *
30 '* X1 VIDEO UTILITY (SHUTTER) V.1 *
40 '*                                *
50 '*    Oct. '83 COPYRIGHT (C)      *
60 '*               by H.ASHIKAGA    *
70 '*                                *
80 '**********************************
90 '
100 WIDTH 40:INIT:PALET:SCREEN 0,0:CLS4
110 DIM A%(126)
120 COLOR 7,0:GOSUB "BOX":CLS4
130 LOCATE 4,10:INPUT"SELECT OPEN (1) OR SHUT (2) ";K
140 IF K<1 OR K>2 THEN 130
150 LOCATE 4,14:INPUT"BACK COLOR ﾃﾞ ｴﾗﾝﾃﾞ ｸﾀﾞｻｲ (1-7) ";C
160 IF C<1 OR C>7 THEN 150
170 COLOR 7,0:CLS
180 LOCATE 7,2:PRINT"♠♠  SELECT MOVEMENT  ♠♠"
190 LOCATE 10,6:PRINT"LEFT TO RIGHT ... 1"
200 LOCATE 10,8:PRINT"RIGHT TO LEFT ... 2"
210 LOCATE 10,10:PRINT"UP TO DOWN    ... 3"
220 LOCATE 10,12:PRINT"DOWN TO UP    ... 4"
230 LOCATE 10,14:PRINT"SCREW         ... 5"
240 LOCATE 10,16:PRINT"AT RANDOM     ... 6"
250 COLOR 4:LOCATE 6,20:INPUT"** HIT 1-6 KEY ";I
260 IF I<1 OR I>6 THEN 180
270 CLS
280 ON I GOSUB "LTOR","LTOR","UTOD","UTOD","SCREW","ATRA"
290 IF INKEY$="" THEN 290
300 PALET:CLS4:END
310 LABEL "BOX"
320 LINE(0,0)-(39,24),PSET,1,BF
330 GET@(0,0)-(39,24),A%,1
340 RETURN
350 LABEL "LTOR"
360 FOR IX=0 TO 319 STEP 40
370 FOR IY=0 TO 199 STEP 25
380 IF K=1 THEN PALET 0,C
390 IF I=1 THEN PUT@(IX,IY)-(IX+39,IY+24),A%,PSET,1 ELSE PUT@(319-39-IX,199-24-I
Y)-(319-IX,199-IY),A%,PSET,1
400 IF K=1 THEN PALET 1,0 ELSE PALET 1,C
410 GOSUB "TIMER"
420 NEXT:NEXT
430 IF K=1 THEN COLOR 7,0 ELSE COLOR 7,C
440 RETURN
450 LABEL "UTOD"
460 FOR IY=0 TO 199 STEP 25
470 FOR IX=0 TO 319 STEP 40
480 IF K=1 THEN PALET 0,C
490 IF I=3 THEN PUT@(IX,IY)-(IX+39,IY+24),A%,PSET,1 ELSE PUT@(319-39-IX,199-24-I
Y)-(319-IX,199-IY),A%,PSET,1
500 IF K=1 THEN PALET 1,0 ELSE PALET 1,C
510 GOSUB "TIMER"
520 NEXT:NEXT
530 IF K=1 THEN COLOR 7,0 ELSE COLOR 7;C
540 RETURN
550 LABEL "SCREW"
560 IX=40*4:IY=25*4
570 FOR ID=0 TO 62
580 READ D:IF D=1 THEN IX=IX+40
590 IF D=2 THEN IX=IX-40
600 IF D=3 THEN IY=IY+25
610 IF D=4 THEN IY=IY-25
620 IF K=1 THEN PALET 0,C
630 PUT@(IX,IY)-(IX+39,IY+24),A%,PSET,1
640 IF K=1 THEN PALET 1,0 ELSE PALET 1,C
650 GOSUB "TIMER"
660 NEXT ID
670 IF K=1 THEN COLOR 7,0 ELSE COLOR 7,C
680 RETURN
690 LABEL "ATRA"
700 IX=INT(RND*8)*40:IY=INT(RND*8)*25
710 IF K=1 THEN PALET 0,C
720 PUT@(IX,IY)-(IX+39,IY+24),A%,PSET,1
730 IF K=1 THEN PALET 1,0 ELSE PALET 1,C
740 GOTO 700
750 LABEL "TIMER"
```

```
760 FOR T=0 TO 100:NEXT
770 RETURN
780 DATA 0,4,2,3,3,1,1,4,4,4,2,2,2,3,3,3,3,1,1,1,1,4,4,4,4,4,2,2,2,2,2
790 DATA 3,3,3,3,3,3,1,1,1,1,1,1,4,4,4,4,4,4,4,2,2,2,2,2,2,2,3,3,3,3,3,3
```

■―❹SCORE WRITER FOR BASEBALL

野球の実況中継のように，子供の少年野球やソフトボール大会，お父さんの草野球の録画画面に，ボールカウントやスコアが表示できたらどんなに素敵な本格的記録ができあがることでしょう。こんな夢を，X1はいとも簡単にかなえてくれます。

プログラムをRUNすると，野球中継の軽快なメロディーとともにタイトルが表示されます。任意のキーを入力すると，チーム名の入力モードになります。自分のチーム名と，対戦相手のチーム名を入れてください。ただし画面にスコアとともに表示されるのは，6文字までです。写真9のようにチーム名が表示されるので，記録用タイトルとして背景にVTRの場面を映し出しながらスーパーインポーズします。

適当な時間記録できたら任意のキーを打ち，写真10のコマンドを表示させます。ここはスーパーインポーズする必要はなく，もちろんVTRテープに記録する必要もありません。コマンドキーを入力すると，写真11のようにチーム名とスコア，ボールカウントが表示されます。Sのキーを押すとストライクがひとつ増え，黄色の丸で表示されます。ストライク3でボールの表示は消え，アウトカウントが1増えます。Bはボールを示し，青い丸で表示されます。ボール4でストライクがクリアされます。Oはアウトカウントで，赤い丸で示されます。アウト3ですべてのボールカウントがクリアされます。自分のチームのスコア，相手チームのスコアは，それぞれY，Tのキーを押すごとに1ずつ増えていきます。訂正が生じたときはCのキーを押し，該当するキーを押していくと，スコアまたはカウントが1ずつ減っていきます。正しい値になったところで，再度Cのキーを押すともとの入力モードに戻ります。

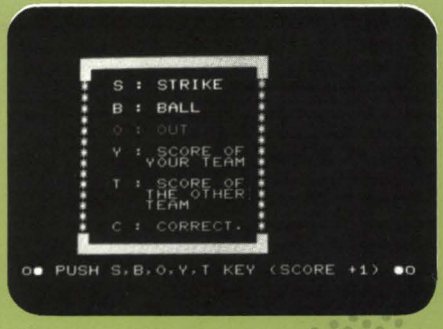
●ワンポイントテクニック

リスト4の730行のテーマ曲を，編集時にアフレコすると効果が上がります。X1の拡大文字機能をふんだんに使い，表示をわかりやすくし，カウントは色分けして一目見て区別できるようにしました。訂正コマンドがあるので，安心して編集できます。310行にINKEY$を使っていますが，背景のVTR画面に応じてタイトルの表示時間は一定しないので，このように任意のキーでいつでも次の場面に移れるようにすると便利です。プログラムを短くするためにキー入力するごとに前画面をクリアしていますが，このとき画面が一瞬まばたくように感じる人は，一工夫してみてください。

また550行から600行のように，CAPSキーがロックしてあってもなくても表示が可能であれば，プログラムを知らない人でもすぐできます。

◀写真9
SCORE WRITER
によるTITLE作製

写真10 ▶
SCORE WRITER
のコマンド表示



リスト4　SCORE WRITER FOR BASEBALL

```
10 '*************************************************
20 '*                                               *
30 '* X1 VIDEO UTILITY (SCORE WRITER FOR BASEBALL) *
40 '*            Ver.2                              *
50 '*     Jan. '84 COPYRIGHT (C) BY H.ASHIKAGA      *
60 '*                                               *
70 '*************************************************
80 '
90 WIDTH40:INIT:PALET:CLS4:GOSUB "TITLE":CLS
100 LOCATE 4,8:INPUT"YOUR TEAM NAME";Y$:PRINT
```

```
110 LOCATE 4,12:INPUT"THE OTHER TEAM'S NAME";O$:CLS
120 CIRCLE(30,100),20,7:CIRCLE(290,100),20,7
130 PAINT(30,100),7,7:PAINT(290,100),7,7
140 CIRCLE(60,100),20,2,1,134,224
150 CIRCLE(60,100),19,2,1,134,224
160 CIRCLE(260,100),20,2,1,314,405
170 CIRCLE(260,100),19,2,1,314,405
180 CIRCLE(0,100),20,2,1,314,405
190 CIRCLE(0,100),19,2,1,314,405
200 CIRCLE(320,100),20,2,1,134,224
210 CIRCLE(320,100),19,2,1,134,224
220 CIRCLE(160,-130),200,2,1,240,300
230 CIRCLE(160,-128),190,2,1,240,300
240 CIRCLE(160,-125),180,6,1,240,300
250 CIRCLE(160,330),200,2,1,60,120
260 CIRCLE(160,328),190,2,1,60,120
270 CIRCLE(160,325),180,6,1,60,120
280 COLOR4:LOCATE 8,10:CSIZE3:PRINT#0,USING"&           &";Y$
290 LOCATE 18,12:PRINT#0"--"
300 LOCATE 16,14:PRINT#0,USING"&           &";O$
310 A$=INKEY$:IF A$="" THEN 310
320 CLS4:PRINT:COLOR 4
330 PRINT"      ██████████████
340 PRINT"      ■            ■
350 PRINT"      *";:COLOR6:PRINT"  S : STRIKE";:COLOR4:PRINT"   *
360 PRINT"      *            *
370 PRINT"      *";:COLOR1:PRINT"  B : BALL  ";:COLOR4:PRINT"   *
380 PRINT"      *            *
390 PRINT"      *";:COLOR2:PRINT"  O : OUT   ";:COLOR4:PRINT"   *
400 PRINT"      *            *
410 PRINT"      *  Y : SCORE OF *
420 PRINT"      *     YOUR TEAM *
430 PRINT"      *            *
440 PRINT"      *  T : SCORE OF *
450 PRINT"      *     THE OTHER *
460 PRINT"      *     TEAM     *
470 PRINT"      *            *
480 PRINT"      *  C : CORRECT. *
490 PRINT"      ■            ■
500 PRINT"      ██████████████
510 PRINT
520 PRINT"○● PUSH S,B,O,Y,T KEY (SCORE +1) ●○"
530 S=0:B=0:O=0:Y=0:T=0:z=1
540 A$=INKEY$
550 IF A$="S" OR A$="s" THEN S=S+Z:GOTO 620
560 IF A$="B" OR A$="b" THEN B=B+Z:GOTO 620
570 IF A$="O" OR A$="o" THEN O=O+Z:S=0:B=0:GOTO 620
580 IF A$="Y" OR A$="y" THEN Y=Y+Z:GOTO 620
590 IF A$="T" OR A$="t" THEN T=T+Z:GOTO 620
600 IF A$="C" OR A$="c" THEN IF Z=-1 THEN Z=+1:GOTO 620 ELSE Z=-1:GOTO 620
610 IF A$="" THEN 540
620 IF S=3 THEN S=0:O=O+1
630 IF B=4 THEN B=0:S=0
640 IF O=3 THEN O=0
650 CLS:COLOR 7:CSIZE1:LOCATE 2,0:PRINT #0,USING"&    & ## - ## &    &";Y$;Y;T;O
$:CSIZE2
660 LOCATE 26,16:COLOR 7:PRINT #0,"S ";:COLOR 6:FOR I=1 TO S:PRINT "●";:NEXT
670 LOCATE 26,18:COLOR 7:PRINT #0,"B ";:COLOR 1:FOR I=1 TO B:PRINT "●";:NEXT
680 LOCATE 26,20:COLOR 7:PRINT #0,"O ";:COLOR 2:FOR I=1 TO O:PRINT "●";:NEXT
690 GOTO 540
700 LABEL "TITLE"
710 CSIZE 3:COLOR 6:LOCATE 4,8:PRINT #0,"VTR SCORE WRITER "
720 LOCATE 8,12:PRINT #0,"FOR BASEBALL"
730 TEMPO 200:MUSIC"V15O4C3-B5C3D5C3-B5C3D5C3-B5C3E1R6E7:V13O6C3-B5C3D5C3-B5C3D5
C3-B5C3E1R6E7"
740 IF INKEY$="" THEN 740
750 RETURN
```

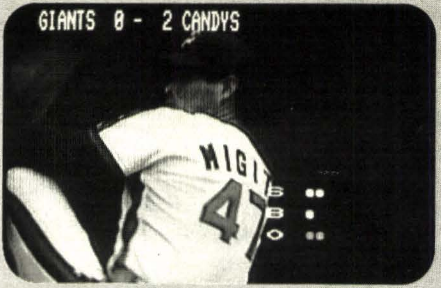


写真11 SCORE WRITER実行例

おわりに

前回と今回の2回にわたってVTRの編集に便利なユーティリティプログラムを紹介しましたが，いかがでしたでしょうか。せっかくのX1のスーパーインポーズ機能です。VTR編集までいかなくても，アンテナをつないで今回のプログラムとスーパーインポーズし，いろいろな遊び方を考えてみてください。次回は基本構成での楽しみ方「TVと遊ぶ方法」を紹介します。どんなものがとび出すかお楽しみに。

プログラムを作るプログラム

ELDER MAN

ちょっと変わったプログラムを作ってみたので紹介したいと思います。プログラムを作るプログラム，自己増殖プログラムというとSFっぽくて魅力を感じないではいられませんネ。X1のHuBASICには多種多様な命令が用意されているため，このような目的を達成できるプログラムに挑戦してみました。

■プログラムの機能について

プログラムを作るプログラムといっても今回のものは，簡単に言ってしまえばプログラムレス言語あるいは簡易言語ということになるでしょうか。X1を電卓の代わりに使おうという人や，簡単な計算のプログラムをいちいち作るのが面倒だという人，またはBASICをよく知らない人には便利なものだと思います。

フローチャートを書いて，順々に入力していけば，プログラムが自動的に作成されるというものです。機能を順に紹介すると，

INPUT（入力）……変数名をアルファベットで入力します。

CAL　（計算）……計算式を変数名＝式という形で入力します。

PRINT（出力）……画面に表示したい変数名を入力します。

この3つの機能だけでも，繰り返し使うような計算には便利です。

たとえば，リンゴが1個200円，ミカンが1個80円として，それぞれ何個か買ったときの合計金額を求めるというのは，

(入力)　RINGO ↵

(入力)　MIKAN ↵

(計算)　ANSWER＝RINGO＊200＋MIKAN＊80 ↵

(出力)　ANSWER ↵

とするだけで，プログラムができて，リンゴの個数，ミカンの個数を聞いてきて，計算し，結果を出力するということができます。計算のところには，何回でもまた複雑な式でも入れることができるので，これだけの機能でも結構便利だと思います。

機能としてはこの他に，

FOR～NEXT（繰り返し）……繰り返しの始めのところで指定すると，繰り返しに使う変数名（単純に繰り返すだけならIか何か適当な変数名でよい）と回数または回数を与える変数や式を入力します。繰り返しの終わるところではNEXTを指定します。

IF～THEN～ELSE（分岐文）……条件を式などで示すことにより，計算方法を変更したりすることができます。分岐する先を示すためには，次に示すLABELの考え方を必要とします。

LABEL（ラベル）……計算の処理工程の区切りにつけたり，分岐文によって示されるジャンプ先を表わすためにLABEL（ラベル）名をアルファベットで入力します。

GOTO（無条件ジャンプ）……計算などで共通に使うようなサブルーチンへ分岐するための命令で，ジャンプ先はLABEL名で示します。

DIM　（配列変数）……多少難しい計算プログラムをも組めるように配列変数を扱えるようにしました。配列変数の使い方については，マニュアルを読んでください。

以上の，INPUT，PRINT，FOR～NEXT，IF～THEN～ELSE，GOTO，DIM文が使用できれば，計算できないものはほとんどないと言ってもいいくらいだと思います。

■プログラムの使い方

実際の使用方法について説明します。まずプログラムをRUNさせます。画面にこのプログラムの使用方法と注意事項が表示されます。注意事項というのは，変数名にはアルファベットを使うこと（2文字目以降は数字でもいい），GOTOとかRUNとかのBASICの予約語は変数名に使用できないということです。また変数名の最後に$をつければ文字変数として，＃をつければ倍精度の数値変数として，％をつければ整数型の変数として扱うことができます。

大文字を使うので，CAPS LOCKしてください。

どれかキーを押すと，モウイチド＝(R)　サイショカラ＝(I)，シュウセイ＝(D)，アンナイ＝(A)，オワリ＝(E)と表示されるので，Iを入力します。プログラム作成のモードとなり，画面では最上行にタイトル，最下行にコマンドの入力案内が表示されます。

D＝配列変数の定義　(DIM)

I＝入力変数名の入力　(INPUT)

C＝計算式の入力　(CAL)

P＝出力変数名の入力　(PRINT)

プログラムの概要

行	内容	行	内容
10～ 70	タイトル	1000～1050	(DIM) 配列変数の定義
80～ 90	BASICのワークエリア確保とエラー発生時処理	1100～1150	(GOTO) 無条件ジャンプ命令処理
100～ 170	初期設定（2回目からは自動削除）	1200～1240	(END) プログラム作成モード終了
180～ 230	作成，新プログラム，案内への分岐	1300～1340	(LABEL) ラベル名の入力処理
	作成モードの画面表示	1400～1810	エラーメッセージ用サブルーチン
240～ 275	新プログラムの行番号セット	1900～1920	新プログラム作成命令
280～ 293	コマンド入力と各処理への分岐	2000～7990	新プログラム作成エリア
300～ 350	(INPUT) 変数名の入力処理	8000～8090	作成，新プログラムの実行などへの分岐
400～ 450	(CAL) 計算式の入力処理	8100～8150	初期設定のやり直し
500～ 550	(PRINT) 出力変数名の入力処理	8200～8310	新プログラムのBASICリスト表示
600～ 670	(FOR) 繰り返しの設定	8400～8520	新プログラムを残して，本体部分は削除
700～ 750	(NEXT) 繰り返しの終わり	9000～9090	新プログラム実行時のエラー発生時処理
800～ 920	(IF) IF～THEN～ELSE処理	9100～9260	このプログラムの使用方法表示

F＝繰り返しの始め (FOR)
N＝繰り返しの終わり (NEXT)
M＝分岐文の作成 (IF～THEN)
G＝無条件ジャンプ (GOTO)
L＝ラベル名の入力 (LABEL)
E＝プログラム作成終了 (END)

Commandとブルーの文字で表示され，カーソルが点滅したらこれらのコマンドを入力します。間違えたら〝／〟を入力すればコマンドはキャンセルされてまたCommandの表示に戻ります。リンゴとミカンの例では，入力のところをI，計算のところをC，出力のところをPとすればよく，最後にEを入力するとRINGO＝と聞いてきます。これはすでにできあがったプログラムが動いているものです。10個と入力してやれば次にMIKAN＝と聞いてきます。同様に10個と入力すれば，計算されてANSWER＝2800と表示されて答が得られます。違う数で再計算のときはRを入力，プログラムを作り直すならばIを入力すればOKです。ここでEを入力するとデキタ プログラム ヲ SAVE シマスカ ? (Y/N) と聞いてくるので，作ったプログラムを保存するならばYを入力してください。このプログラムを作るプログラムは消えて，作ったプログラムだけが残ります。リストを出せば，2000行から作られたプログラムリストが表示されます。

Dを入力すると，作ったプログラムリストが表示されます。ここでは，不都合なところをBASICプログラム上で修正することができます（BASICがわからない人は，もう一度初めからやり直すしかありませんネ，あしからず）。

■計算例

1からNまでの数の総和を求めるプログラムの作成例を紹介します。

コマンド

(I) N (数の入力)
(F) I (Iという変数，1ずつ増加)
N (N回繰り返す)
(C) ANS＝ANS＋I (Nまで加える)
(N) (繰り返し終わり)
(P) ANS (答を表示)
(E) (終了)

これだけの操作で

```
2010  INPUT "N=", N
2020  FOR I=1 TO N
2030  ANS=ANS+I
2040  NEXT
2050  PRINT "ANS=" ;ANS
```

というプログラムが作られます。複雑なプログラムになると，ラベル名をつけ忘れることがあるので，その場合は，BASIC上でラベル名を追加すればよいと思います。

■プログラムの構造

BASICでBASICのプログラムを作るということは，これまでなかなかできないことでした。X1で可能になったのはKEY0，という命令のおかげです。プログラムリストをみていただければわかるように，多用しています。160行にKEY0，"RUN"＋CHR$(13) とあるのが最初で，何をしているかというと，プログラムを初めてRUNしたときの初期値のセットを行なっています。170行にDELETE 100－170とあるように，ここまでプログラムが走ってくると，リストの100行から170行までが削除されてしまいます。KEY0，文がないとプログラムはここでSTOPしてしまいますが，KEY0，文のあとに"RUN"＋CHR$(13) とあるため，再びプログラムは頭からスタートします。2度目はもう100行から170行まではなくなっているので180行以上の行へプログラムは進みます。

プログラムを作る部分は，1900行から1920行までで，やはりKEY0，のあとに"RUN"＋CHR$(13) というのがあります。AAAA$というのが新しく作るプログラムの部分で行番号とプログラムが代入されています。COLOR0としてあるためCRT上では見えませんが，プログラムリストを登録させるときに↵を押しているのと同じことをここで行なっています。新しいプログラムの登録が終わると再びスタートして，何回も同じ作業を繰り返して，1行ずつ新しいプログラムを作成していきます。

プログラムをRUNするとすべての変数がクリアされてしまうため，新しく作る行番号，作成モードと実行モードとの区別，FOR～NEXTによる繰り返しループの完結確認のためのパラメータなどについては，BASIC TEXTエリア外におく必要があります。80行でLIMIT &HFEF0としているのはこのためです。

■さいごに

プログラムを作るプログラムがもっと高級なことをできれば，SFの世界でのことがあるいは現実になるかも知れません。どなたか，独創的なアイデアでプログラムを作るプログラムを発展させてみませんか。

リスト

```
10 '=============================================================
20 '
30 ' AUTO PROGRAM (Proto Type 4)  Copyright 1984  by  ELDER MAN
40 '
50 '                    ONLY CALCULATION VERSION
60 '
70 '=============================================================
80 LIMIT &HFEF0
90 ON ERROR GOTO 9000
100 '
110 '----- INITIALIZE -----
120 '
130 WIDTH 80:COLOR0
140 DEF KEY 5,"GOTO 8000"+CHR$(13)
150 POKE &HFEF0,0:POKE &HFEF1,3:POKE &HFEF2,128
160 KEY0,"RUN"+CHR$(13)
170 DELETE 100-170
180 '
190 '----- TITLE & GUIDE -----
200 '
203 ON PEEK(&HFEF1) GOTO 205,2000,9100
205 IF PEEK(&HFEF1)<>0 THEN 240 ELSE POKE &HFEF1,1
210 CONSOLE:CLS:COLOR5:LOCATE0,0:PRINT"■■■■■■■■  AUTO PROGRAM (Proto Type 4)  C
opyright 1984  by  A.SHIRAI  ■■■■■■■■"
220 LOCATE0,24:PRINT"D=DIM◢ I=INPUT◢ C=CAL.◢ P=PRINT◢ F=FOR◢ N=NEXT◢ M=IF◢ G=GOT
O◢ L=LABEL◢ E=END◢";
230 CONSOLE 1,23
240 '
250 '----- POINTER SET -----
260 '
270 IIII=PEEK(&HFEF0)
275 IIII=IIII+1:POKE &HFEF0,IIII
278 '
279 '----- COMMAND -----
280 '
281 COLOR1:PRINT CHR$(30);"Command ";
282 HHHH$=INKEY$(1)
283 IF HHHH$="I" THEN 300
284 IF HHHH$="C" THEN 400
285 IF HHHH$="P" THEN 500
286 IF HHHH$="F" THEN 600
287 IF HHHH$="N" THEN 700
288 IF HHHH$="M" THEN 800
289 IF HHHH$="D" THEN 1000
291 IF HHHH$="G" THEN 1100
292 IF HHHH$="E" THEN 1200
293 IF HHHH$="L" THEN 1300 ELSE 282
297 '
298 '----- INPUT VAR. NAME -----
299 '
300 COLOR4:PRINT"ﾍﾝｽｳﾒｲ ｦ ﾆｭｳﾘｮｸ (INPUT)":BEEP
310 LINPUT "",HHHH$
320 IF HHHH$="/" THEN 281
330 IF ASC(LEFT$(HHHH$,1))<65 OR ASC(LEFT$(HHHH$,1))>90 GOSUB 1400:GOTO 300
340 AAAA$=STR$(IIII*10+2000)+" INPUT "+CHR$(34)+HHHH$+CHR$(61,34,44)+HHHH$
350 GOTO 1900
397 '
398 '----- INPUT How to Calculation -----
399 '
400 COLOR6:PRINT"ｹｲｻﾝ ｼｷ ｦ ﾆｭｳﾘｮｸ (CAL.)":BEEP
410 LINPUT "",HHHH$
420 IF HHHH$="/" THEN 281
425 IF INSTR(HHHH$,"=")=0 GOSUB 1700:GOTO 281
430 IF ASC(LEFT$(HHHH$,1))<65 OR ASC(LEFT$(HHHH$,1))>90 GOSUB 1400:GOTO 400
440 AAAA$=STR$(IIII*10+2000)+" "+HHHH$
450 GOTO 1900
497 '
498 '----- INPUT PRINT VAR.NAME -----
499 '
500 COLOR3:PRINT"ﾋｮｳｼﾞ ﾍﾝｽｳﾒｲ ｦ ﾆｭｳﾘｮｸ (PRINT)":BEEP
510 LINPUT "",HHHH$
520 IF HHHH$="/" THEN 281
530 IF ASC(LEFT$(HHHH$,1))<65 OR ASC(LEFT$(HHHH$,1))>90 GOSUB 1400:GOTO 500
540 AAAA$=STR$(IIII*10+2000)+" PRINT "+CHR$(34)+HHHH$+CHR$(61,34,59)+HHHH$
550 GOTO 1900
597 '
598 '----- INPUT LOOP NO. -----
599 '
600 COLOR7:BEEP:PRINT:PRINT"ｸﾘｶｴｼ ﾉ ﾍﾝｽｳ ﾒｲ ｶ (FOR - ) = ":LINPUT HHHH$
610 IF HHHH$="/" THEN 281
620 IF ASC(LEFT$(HHHH$,1))<65 OR ASC(LEFT$(HHHH$,1))>90 GOSUB 1400:GOTO 600
630 BEEP:PRINT"ｸﾘｶｴｼ ﾉ ｶｽﾞ ｶ (TO -) = ":LINPUT LLLL$
640 IF LLLL$="/" THEN 281
650 MMMM=PEEK(&HFEF2):MMMM=MMMM+1:POKE &HFEF2,MMMM
```

```
660 AAAA$=STR$(IIII*10+2000)+" FOR "+HHHH$+CHR$(61,49)+" TO "+LLLL$
670 GOTO 1900
697 '
698 '----- LOOP END (NEXT) -----
699 '
700 COLOR7:BEEP:INPUT"ﾙｰﾌﾟ ｵﾜﾘ (NEXT) ? (Y or /)",HHHH$
710 IF HHHH$="/" THEN PRINT:GOTO 281
720 IF PEEK(&HFEF2)=128 GOSUB 1600:GOTO 281
730 MMMM=PEEK(&HFEF2):MMMM=MMMM-1:POKE &HFEF2,MMMM
740 AAAA$=STR$(IIII*10+2000)+" NEXT"
750 GOTO 1900
797 '
798 '----- IF - THEN - ELSE -----
799 '
800 COLOR5:PRINT:PRINT"ﾊﾝﾃｲ ﾌﾞﾝｷ (IF - THEN - ELSE -)":BEEP
810 PRINT"ﾊﾝﾃｲ ｼﾞｮｳｹﾝ ; ":LINPUT HHHH$
820 IF HHHH$="/" THEN 281
830 IF ASC(LEFT$(HHHH$,1))<65 OR ASC(LEFT$(HHHH$,1))>90 GOSUB 1400:GOTO 800
840 COLOR3:PRINT"ﾌﾞﾝｷ ｻｷ ﾉ LABEL ﾒｲ ﾏﾀﾊ ｹｲｻﾝ ｱｲﾖｳ":BEEP
850 PRINT"LABEL ﾒｲ ﾉ ﾄｷ ﾊ ";CHR$(34,32,34);" ﾃﾞ ｶｺﾝﾃﾞｸﾀﾞｻｲ"
860 PRINT"YES ﾉ ﾄｷ ; ":LINPUT "",LLLL$
870 IF LLLL$="/" THEN 281
880 PRINT"NO  ﾉ ﾄｷ ( ﾌﾖｳ ﾅﾗ N ｦ ﾆｭｳﾘｮｸ ); ":LINPUT KKKK$
890 IF KKKK$="/" THEN 281
900 IF KKKK$="N" THEN KKKK$="" ELSE KKKK$=" ELSE "+KKKK$
910 AAAA$=STR$(IIII*10+2000)+" IF "+HHHH$+" THEN "+LLLL$+KKKK$
920 GOTO 1900
997 '
998 '----- DIM -----
999 '
1000 COLOR7:PRINT"ﾊｲﾚﾂ ﾍﾝｽｳ ﾉ ﾃｲｷﾞ"
1010 PRINT"ｼｮｼｷ = ﾍﾝｽｳﾒｲ(N),ﾍﾝｽｳﾒｲ(L,M).....":BEEP
1020 LINPUT "",HHHH$
1030 IF HHHH$="/" THEN 281
1035 IF INSTR(HHHH$,"(")=0 OR INSTR(HHHH$,")")=0 GOSUB 1800:GOTO 1000
1040 IF ASC(LEFT$(HHHH$,1))<65 OR ASC(LEFT$(HHHH$,1))>90 GOSUB1400:GOTO 1000
1050 AAAA$=STR$(IIII*10+2000)+" DIM "+HHHH$:GOTO 1900
1097 '
1098 '----- GOTO -----
1099 '
1100 COLOR6:BEEP:PRINT"ﾑｼﾞｮｳｹﾝ ｼﾞｬﾝﾌﾟ (GOTO)"
1110 INPUT"LABEL NAME = ";HHHH$
1120 IF HHHH$="/" THEN PRINT:GOTO 281
1130 IF ASC(LEFT$(HHHH$,1))<65 OR ASC(LEFT$(HHHH$,1))>90 GOSUB 1500:GOTO 1100
1140 AAAA$=STR$(IIII*10+2000)+" GOTO "+CHR$(34)+HHHH$+CHR$(34)
1150 GOTO 1900
1197 '
1198 '----- END & CAL.START -----
1199 '
1200 IF PEEK(&HFEF2)=128 THEN 1230
1210 COLOR2:PRINT"ERROR !!":BEEP
1220 PRINT"ｸﾘｶｴｼ ﾉ ｵﾜﾘ ｶﾞ ﾅｲ (NEXT ｶﾞ ﾅｲ) !":PRINT:GOTO 281
1230 AAAA$=STR$(IIII*10+2000)+" GOTO 8000":POKE &HFEF1,2
1240 GOTO 1900
1297 '
1298 '----- LABEL -----
1299 '
1300 COLOR5:BEEP:INPUT"LABEL ﾉ ﾅﾏｴ = ",HHHH$
1310 IF HHHH$="/" THEN PRINT:GOTO 281
1320 IF ASC(LEFT$(HHHH$,1))<65 OR ASC(LEFT$(HHHH$,1))>90 GOSUB 1500:GOTO 1300
1330 AAAA$=STR$(IIII*10+2000)+" LABEL"+CHR$(34)+HHHH$+CHR$(34)
1340 GOTO 1900
1397 '
1398 '----- ERROR MESSAGE -----
1399 '
1400 COLOR2:PRINT "ERROR !!":BEEP
1410 PRINT "ﾍﾝｽｳﾒｲ ﾊ ｱﾙﾌｧﾍﾞｯﾄ ﾃﾞ !!":RETURN
1500 COLOR2:PRINT "ERROR !!":BEEP
1510 PRINT "LABEL ﾒｲ ﾊ ｱﾙﾌｧﾍﾞｯﾄ ﾃﾞ !!":RETURN
1600 COLOR2:PRINT "ERROR !!":BEEP
1610 PRINT"ﾀｲｵｳ ｽﾙ FOR ｶﾞ ｱﾘﾏｾﾝ !!":PRINT:RETURN
1700 COLOR2:PRINT "ERROR !!":BEEP
1710 PRINT"ｼｷ ﾉ ｹｲｻﾝ ｶﾞ ﾃﾞｷﾏｾﾝ !!":PRINT:RETURN
1800 COLOR2:PRINT "ERROR !!":BEEP
1810 PRINT"ｼｮｼｷ ｶﾞ ﾁｶﾞｯﾃ ｲﾏｽ !!":RETURN
1897 '
1898 '----- MAKE PROGRAM -----
1899 '
1900 COLOR 0
1910 KEY0,AAAA$+CHR$(13,30,30,26)+"RUN"+CHR$(13)
1920 END
1960 '
1970 '----- NEW PROGRAM AREA -----
1980 '
2000 COLOR7:CONSOLE:CLS
2002 PRINT"жжжж  Auto Program Calculation   RUN & RESULT  by  Proto 4  жжжж"
```

```
2005 CONSOLE1,23:COLOR7:PRINT
7997 '
7998 '----- RETRY ? -----
7999 '
8000 BEEP:BEEP:BEEP
8030 COLOR4:PRINT "Input ! ﾓｳｲﾁﾄﾞ=(R),ｻｲｼｮｶﾗ=(I),ｼｭｳｾｲ=(D),ｱﾝﾅｲ=(A),ｵﾜﾘ=(E)"
8040 HHHH$=INKEY$
8050 IF HHHH$="A" THEN 9100
8060 IF HHHH$="R" THEN CLR:GOTO 2000
8070 IF HHHH$="I" THEN 8100
8080 IF HHHH$="D" THEN 8200
8090 IF HHHH$="E" THEN 8400 ELSE 8040
8097 '
8098 '----- INITIALIZE -----
8099 '
8100 POKE&HFEF0,0:POKE &HFEF1,0:POKE &HFEF2,128
8140 KEY0,"GOTO 10"+CHR$(13)
8150 DELETE 2010-7990
8197 '
8198 '----- PROGRAM DEBUG -----
8199 '
8200 CONSOLE:CLS:COLOR7:PRINT"■■■■■    BASIC Program Debug   by  Screen Edit   ■
■■■■"
8240 BEEP
8250 COLOR5:PRINT"  Program ｦ ｼｭｳｾｲ ｼﾃｸﾀﾞｻｲ !"
8260 COLOR5:PRINT"  Screen ﾃﾞ ﾅｵｼﾃ CR ｷｰ ｦ ｵｼﾃ ｸﾀﾞｻｲ !"
8270 COLOR3:PRINT"  Program List ｶﾞ ﾅｶﾞｲ ﾊﾞｱｲ ﾊ BREAK ｷｰ ﾃﾞ ﾄﾏﾘﾏｽ｡"
8280 COLOR3:PRINT"  ﾂｷﾞﾉ List ﾆ ｽｽﾑ ﾄｷ 、ｽﾍﾟﾃ ｵﾜｯﾀ ﾄｷ ﾊ F5 ｷｰ ｦ ｵｼﾃ ｸﾀﾞｻｲ!"
8290 COLOR2:PRINT"  ﾄﾞﾚｶ ｷｰ ｦ ｵｼﾃ ｸﾀﾞｻｲ !":BEEP
8300 IF INKEY$="" THEN 8300
8310 COLOR7:LIST 2010-7990:END
8397 '
8398 '----- END -----
8399 '
8400 COLOR5:PRINT"ﾃﾞｷﾀ ﾌﾟﾛｸﾞﾗﾑ ｦ SAVE ｼﾏｽｶ ? (Y/N)"
8410 HHHH$=INKEY$:IF HHHH$="" THEN 8410
8420 IF HHHH$="N" THEN END
8425 COLOR7:PRINT"ﾂｸｯﾀ ﾌﾟﾛｸﾞﾗﾑ ﾀﾞｹ ｶﾞ ﾉｺﾘﾏｽ !":BEEP:PAUSE 50
8430 KEY0,"GOTO 8450"+CHR$(13)
8440 DELETE 7997-8440
8450 KEY0,"GOTO 8470"+CHR$(13)
8460 DELETE 10-2005
8470 KEY0,"GOTO 8490"+CHR$(13)
8480 DELETE 8450-8480
8490 KEY0,"8000 END"+CHR$(13)+"GOTO 8495"+CHR$(13)
8492 DELETE 8490-8492
8495 KEY0,"2000 '----- NEW PROGRAM -----"+CHR$(13)
8500 DEF KEY 5,"RUN"+CHR$(13)
8510 CONSOLE:CLS
8520 DELETE 8490-
8997 '
8998 '----- ERRROR MESSAGE -----
8999 '
9000 COLOR1:PRINT:PRINT"ERROR ｶﾞ ｱﾘﾏｽ｡ ﾌﾟﾛｸﾞﾗﾑ ｦ ﾐﾅｵｼﾃ ｸﾀﾞｻｲ !"
9010 IF ERR=11 THEN COLOR3:PRINT"ｼｷ ﾉ ｹｲｻﾝ ﾃﾞ 0 ｶﾞ ﾌﾞﾝﾎﾞ ﾆ ｱﾘﾏｾﾝｶ ?":GOTO 9060
9020 IF ERR=8 THEN COLOR4:PRINT"LABEL ﾊ ﾀﾀﾞｼｸ ﾂｹﾗﾚﾃ ｲﾏｽｶ ?":GOTO 9060
9030 IF ERR=9 THEN COLOR5:PRINT"ﾊｲﾚﾂ ﾍﾝｽｳ ﾉ ﾃｲｷﾞ ｦ ﾅｵｼﾃ ｸﾀﾞｻｲ !":GOTO 9060
9040 IF ERR=13 THEN COLOR7:PRINT"ﾓｼﾞﾍﾝｽｳ ﾄ ｽｳﾁﾍﾝｽｳ ﾉ ｸﾍﾞﾂ ｦ ｼﾃｸﾀﾞｻｲ｡":GOTO 9060
9045 IF ERR=2 THEN COLOR5:PRINT"ｼｷ ﾉ ｶﾀﾁ ｶﾞ ﾁｶﾞｯﾃ ｲﾏｾﾝｶ ?"
9050 COLOR6:PRINT"ﾍﾝｽｳﾒｲ ﾆ BASIC ﾉ ﾖﾔｸｺﾞ ﾊ ｱﾘﾏｾﾝｶ ? "
9060 COLOR2:PRINT"ﾄﾞﾚｶ ｷｰ ｦ ｵｼﾃ ｸﾀﾞｻｲ !":BEEP
9090 HHHH$=INKEY$:IF HHHH$="" THEN 9090 ELSE RESUME 8000
9097 '
9098 '----- GUIDE -----
9099 '
9100 CONSOLE:CLS:COLOR5:PRINT"■■■■■  AUTO  PROGRAMMING  GUIDE  for  Proto Type 4
   by  A.SHIRAI  ■■■■■":BEEP
9110 COLOR7:PRINT:PRINT"ｺﾚﾊ BASIC ﾌﾟﾛｸﾞﾗﾑ ｦ ﾂｸﾙ ﾌﾟﾛｸﾞﾗﾑ ﾃﾞｽ｡"
9120 COLOR5:PRINT:PRINT"BASIC ﾉ INPUT,PRINT,FOR-NEXT,IF-THEN-ELSE,GOTO ﾉ ｽﾃｰﾄﾒﾝﾄ
ｦ"
9130 PRINT"ﾂｶｯﾃ ﾌﾟﾛｸﾞﾗﾑ ｦ ﾂｸﾘﾏｽ｡"
9140 PRINT"GOTO ﾔ IF-THEN-ELSE ﾃﾞﾉ ｼﾞｬﾝﾌﾟ ｻｷ ﾊ LABEL ﾒｲ ｦ ﾂｹﾃ ｼﾒｼﾏｽ｡"
9150 COLOR4:PRINT:PRINT"ﾍﾝｽｳﾒｲ ﾆﾊ BASIC ﾉ ﾖﾔｸｺﾞ ｦ ﾉｿﾞｲﾀ ｱﾙﾌｧﾍﾞｯﾄ ｦ ﾂｶｲﾏｽ｡"
9160 PRINT"ﾏﾀ、ﾍﾝｽｳﾒｲ ﾉ ｱﾄﾆ $,#,%,ｦ ﾂｹﾙ ｺﾄｶﾞﾃﾞｷﾏｽ｡"
9170 PRINT"      $=ﾓｼﾞﾍﾝｽｳ、 #=ﾊﾞｲｾｲﾄﾞ、 %=ｾｲｽｳ"
9180 PRINT"CAL(ｹｲｻﾝ) ﾊ    ﾍﾝｽｳ = ｼｷ    ﾉ ｶﾀﾁ ﾃﾞ ﾆｭｳﾘｮｸ ｼﾏｽ｡"
9190 COLOR1:PRINT:PRINT"Command";:COLOR5:PRINT" ﾉ ｱﾄﾆ  I(INPUT),C(CAL),P(PRINT),
F(FOR),N(NEXT),M(IF),G(GOTO),L(LABEL)
9195 PRINT"          E(END) ｦ ﾆｭｳﾘｮｸ ｼﾃｸﾀﾞｻｲ｡"
9200 COLOR1:PRINT"Command";:COLOR5:PRINT" ﾉ ﾄﾘｹｼ ﾊ  ﾉ ｷｰ ｦ ﾆｭﾘｮｸ ｼﾃｸﾀﾞｻｲ"
9210 COLOR4:PRINT:PRINT"ﾃﾞｷｱｶﾞｯﾀ ﾌﾟﾛｸﾞﾗﾑ ﾊ BASIC TEXT ｼﾞｮｳ ﾃﾞ ｼｭｳｾｲ ｼﾃｸﾀﾞｻｲ｡"
9220 PRINT"ﾌﾟﾛｸﾞﾗﾑ ﾊ 2010 ｷﾞｮｳ ｶﾗ ﾂｸﾗﾚﾏｽ｡ 2010 ｷﾞｮｳ ｶﾗ ｱﾄ ｦ ｼｭｳｾｲ ｼﾃｸﾀﾞｻｲ｡"
9230 COLOR3:PRINT:PRINT"CAPS LOCK ｼﾃ ｸﾀﾞｻｲ"
9250 COLOR2:PRINT:PRINT"ﾖｳｲ ｶﾞ ﾖｹﾚﾊﾞ ﾄﾞﾚｶ ｷｰ ｦ ｵｼﾃ ｸﾀﾞｻｲ !":BEEP
9260 IF INKEY$="" THEN 9260 ELSE 8000
```

Oh! ショートプログラム

X1

ファンクションチェンジツール……………香川　次郎

HuBASICのファンクションキーを好みのものに変更したあなた独自の BASIC TAPE を作るプログラムです。

画面に表示される，赤い囲みの中を好きなものに変えていってください。

ちなみに私の作った例は，

[F1] BASIC領域を最大にして，プログラムをLOAD，RUNします（マシン語プログラムを使ったあとでも Out of memory error が出ません）。

[F4] 文字などを初期化し，80字モードでリストを出します。

[F7][F9] インサートモードになっていますので，そのまま数値をキーインできます。

[F10] プログラムをSAVEするとき，

SAVE"ファイルネーム"+[F10]

とすれば，自動的に2回SAVEして，最初のプログラムをベリファイします。

㊟[F4]，[F5]は，シフトクリアキーが入っているので，たとえリストの行中にカーソルがあってもすぐに実行できます。

リスト

```
1 '■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
2 '■                             ■
3 '■  FUNCTION KEY CHANGE PROGRAM ■
4 '■                             ■
5 '■   '84.2.12   by Jiro Kagawa  ■
6 '■                             ■
7 '■    Yamaguchi Rousai Hospital ■
8 '■                             ■
9 '■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■
10 CLEAR &HFC00
20 INIT:WIDTH 40:CLS4
30 REPEAT OFF:KBUF ON:DEFSTR A:ON KEY GOSUB 480
40 POKE &H1A2,&HB7
50 FORI=0 TO 80
60 READ A:B=VAL("&H"+A):POKE &HFE00+I,B
70 C=C+B
80 NEXT
90 FORI=0 TO 159
100 POKE &HFC42+I,0
110 NEXT
120 IF C<>8794 THEN PRINT"DATA CHECK SUM ERROR":REPEAT ON:EDIT 660
130 A=CHR$(34):PRINT"      FUNCTION CHANGE PROGRAM"
140 PRINTSPC(13)"HuBASIC CZ8CB0X"
150 PRINT#0"       KEY 1,"+A+"LIM.&HFEFF:R."+CHR$(34,13,34)
160 PRINT#0"       KEY 2,"+A+"LOCATE /////////"+A
170 PRINT#0"       KEY 3,"+A+"GOSUB //////////"+A
180 PRINT#0"       KEY 4,"+CHR$(34,4,12)+"WIDTH80:LIST"+CHR$(13,34)
190 PRINT#0"       KEY 5,"+CHR$(34,12)+"RUN"+CHR$(13)+"///////////"+A
200 PRINT#0"       KEY 6,"+A+"LIM.&HFEFF:LO."+CHR$(13,34)
210 PRINT#0"       KEY 7,"+A+"FOR I= TO"+CHR$(29,29,29,1)+"//"+A
220 PRINT#0"       KEY 8,"+A+"REPEAT ON"+CHR$(5,13)+"////"+A
230 PRINT#0"       KEY 9,"+A+"LINE()-("+CHR$(29,29,29,1)+"///"+A
240 PRINT#0"       KEY10,"+CHR$(34,13,30,30,30,13)+"AP.-2"+CHR$(13)+"VE."+CHR$(13
,34)
250 COLOR2:LINE(103,15)-(103,96)-(224,96)-(224,7)-(215,7)-(215,15)-(103,15)
260 COLOR3:PRINT"ｱｶｲ ﾜｸ ﾉ ﾅｶ ｦ ｶｷｶｴﾃ ｸﾀﾞｻｲ｡"
270 PRINT"ｳｼﾛ ﾉ ﾖﾊｸ ﾊ ﾉ ﾃﾞ ｳﾒﾃ ｸﾀﾞｻｲ｡ "
275 COLOR6:PRINT"CZ8CB0X ﾉ X ﾊ ｺﾝﾄﾛｰﾙ ｷｬﾗ ﾊ ﾑｼ ｼﾃ 2 ﾄ ﾅﾘﾏｽ｡"
280 COLOR5:PRINT#0"ｺﾝﾄﾛｰﾙ ｷｬﾗｸﾀ "+CHR$(28,32,29,32,30,32,31)+" ﾊ ESCｷｰ ｦ ｲﾁﾄﾞ"
290 PRINT#0"ｵｼﾀｱﾄﾃﾞ､ "+CHR$(13)+" ﾔ "+CHR$(12)+" ﾊ RET. ﾔ CTRL+L ｦ ｵｽ｡"
300 COLOR6:PRINT:PRINT"NEW BASIC ﾊ ESC+RET.ｷｰ ﾅﾄﾞﾃﾞ ﾋｮｳｼﾞﾃﾞｷﾏｽ｡"
310 COLOR4:PRINT#0"PRINT"+CHR$(34,&H9A,32,&H97,29,29,29,31,31,&H99,32,&H98,34)+"
   ﾉ ﾖｳﾆ ﾌﾟﾛｸﾞﾗﾑ ﾉ ﾅｶﾃﾞ ﾂｶｴﾊﾞ ﾒﾓﾘｰ ﾉ ｾﾂﾔｸ ﾆ ﾅﾘﾏｽ｡"
```

```
320 COLOR2:PRINT:PRINT"ｵﾜｯﾀﾗ F-1 ｷｰ + SPACE ｷｰ ﾗ ｵｼﾃｸﾀﾞｻｲ。";
330 X=13:Y=2:COLOR 7
340 LOCATE X,Y:B=ASC(INKEY$(1))
350 IF B=27 THEN B=ASC(INKEY$(1)):GOTO 440
360 IF B=28 THEN X=X+1:D=1
370 IF B=29 THEN X=X-1:D=1
380 IF B=30 THEN Y=Y-1:D=1
390 IF B=31 THEN Y=Y+1:D=1
400 IF X=27 AND Y=1 THEN 440
410 IF X<13 THEN X=13
420 IF X>27 THEN X=13:Y=Y+1
430 IF Y<2 OR Y>11 THEN Y=2
440 C=ASC(CHARACTER$(X,Y))
450 LOCATE X,Y:IF D=1 THEN D=0:PRINT#0 CHR$(C):X=X-1 ELSE PRINT#0 CHR$(B)
460 X=X+1:IF X>27 THEN X=13:Y=Y+1 IF Y>11 THEN Y=2
470 GOTO 340
480 FOR I=1 TO 10
490 FOR J=1 TO 15
500 B=ASC(CHARACTER$(12+J,1+I))
510 IF B=47 THEN POKE &HFC32+16*I,J-1:J=16:GOTO540
520 POKE &HFC32+16*I+J,B
530 IF J=15 THEN POKE &HFC32+16*I,15
540 NEXT:NEXT
550 C=ASC(CHARACTER$(27,1)):IF C<33 THEN C=50
560 POKE &HFE24,C:POKE &HFE29,C
570 EJECT:CLS4:LOCATE 8,12:PRINT"SET BASIC CZ-8CB01 TAPE"
580 IF CMT(1)=0 THEN 580
590 CLS:LOCATE 2,4:PRINT"TAPE ｶﾞ ﾄﾏｯﾀﾗ NEW TAPE ﾜ SET ｼ"
600 LOCATE 7,6:PRINT"SPACE KEY ﾗ ｵｼﾃｸﾀﾞｻｲ"
610 LOCATE 2,10:PRINT"ｶｰｿﾙ ｶﾞ ﾃﾞﾀﾗ ｵﾌﾘﾃﾞｽ          "
620 LOCATE 2,14:PRINT"NEW TAPE ﾗ ｲﾚﾃ SPACE KEY ﾗ ｵｼﾃｸﾀﾞｻｲ。"
630 LOCATE 2,18:PRINT"ﾅﾝﾄﾞ ﾓ COPY ﾃﾞｷﾏｽ"
640 LOCATE 2,22:PRINT"ｵﾜｯﾀﾗ RESET ﾎﾞﾀﾝ ﾗ ｵｼﾃｸﾀﾞｻｲ。"
650 CALL &HFE00
660 DATA 21,60,FE,01,20,00,CD,41,00,2A,74,FE,ED,4B,72,FE,CD,44,00,11,42,0F,21,42
,FC,01,A0,00,ED,B0,3E,B7,32,A2,01,3E,E3,32,6D,FE,3E,E3,32,9A,9D,3E,01,CD,1B,00,F
E,20,20,F7,21,60,FE,01,20,00,CD,3B,00,2A,74,FE,ED,4B,72,FE,CD,3E,00,3E,01,CD,EC,
0D,C3,2D,FE
```

MZ-2200/2000/80B

SCREEN Maker

……角田　裕一

MZのシャープBASICにはKEY0の機能がないため，スクリーンエディットに頼ってきました。このプログラムは，ゲームのデモなどの画面を簡単に作るためのプログラムです。

RUNすると，使用行数を聞いてきます。1～23の間で指定してください。入力すると画面がクリアされ，〝↑↑↑……〟が現われます。そこで，〝↑〟の上までを使って，自由に画面を作ってください。ただし，2つだけルールがあります。

- 書き終わるまで決して[CR]を押さない。
- 書き終わったら，何か書いてある行で[CR]を押す。

以上はINPUT文を使っているため，しかたがないように思われます。

さて，入力が終わり[CR]を押すと，始める行番号を聞いてきます。行番号を入力するとすべての行に番号が付いてリストが出てきますので，最後の行まで[CR]を押していってください。80字モードの画面が40字モードに変わればめでたく完成です。

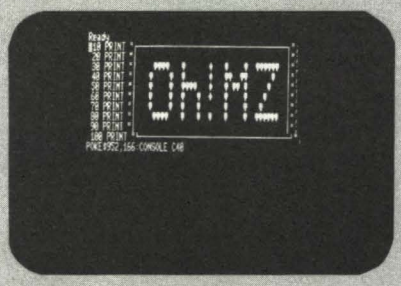
最後に，本プログラムを入力する前に，

POKE$23,5：POKE$24,6[CR]

を入力しておくと，テンキーの[00]が，[H][C]に変わり，打ち込み時に便利です。

リスト

```
10 DEF KEY(5)=POKE$952,166:CONSOLE 50,24,C40
20 PRINT "** SCREEN Maker **":PRINT
30 INPUT "How many lines do you use? (1~23)";LI:LI=LI-1
40 IF (LI<0)+(LI>22) THEN 30
50 PRINT"@":CURSOR 0,LI+1:PRINT STRING$("↑",40);
60 INPUT"ﾛ";A$
70 DIM A$(LI)
80 FOR Y=0 TO LI:FOR X=0 TO 39
90 A$=CHARACTER$(X,Y):IF ASC(A$)<30 THEN A$=" "
100 A$(Y)=A$(Y)+A$:NEXT X:USR($F14):NEXT Y
110 CONSOLE C80
120 INPUT"START LINE : ";A:PRINT"@"
130 FOR I=0 TO LI
140 PRINT A;" PRINT ";CHR$(34);A$(I);CHR$(34);";"
150 A=A+10:NEXT:PRINT"POKE$952,166:CONSOLE C40ﾛ";:POKE$952,166:NEW
```



MZ-80K/C/1200

PASCAL SP-4010用テンキー

…駄本 浩介

PASCAL SP-4010のデータ入力用にと，MZ-80K/C/1200の右側のグラフィックキーをテンキーに変更してみました。ただし，整数型のみで，負の数も使えます。

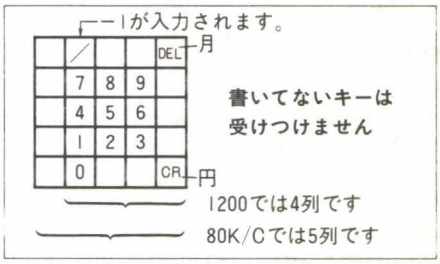
◦利用法

この関数を使えば，PASCAL SP-4010で作ったプログラムの中で，整数型変数Nを使って，

N：TENKEY；

と書いてやれば，Nにテンキーを使って，また通常の入力でも整数データを入力できます。

キーの割り付けは次のとおりです。



◦プログラムの説明

このプログラムは2つの関数からなっています。

KEYINはキー1文字を読みとるもので，6～12行目で，ハートのカーソルを点滅させてキーを読みとります。13行目以下は，受け付けるキーをORでつないでいます。24行目はキーを読んだとき，ハートが出ている場合もあるので消しています。

TENKEYは，KEYINによって読み込んだデータをもとに数値を作ります。

／や」（最上段右から4列目）を入力すると－1になります。

41行目のCINというのは，現在カーソルのある文字のコード（CINコード，マニュアルについている）を読みます。42はマイナス「－」のコードです。

78行目以降はこの関数テストのためのメインで，無限ループ（3＞5になることはないから）になっています。この行以降にあなたのメインプログラムを入力していってください。

リスト

```
 0.VAR N:INTEGER;
 1.%-------------------------%
 2.FUNCTION KEYIN:CHAR;
 3.VAR K:CHAR;
 4.    T:INTEGER;
 5.BEGIN
 6. T:=0;
 7. REPEAT
 8.   IF T<4 THEN WRITE("♥■")
 9.          ELSE WRITE(" ■");
10.   T:=(T+1)MOD 8;
11.   K:=KEY;
12. UNTIL
13.  (K=' ▗')OR(K=' ▏')OR(K='▄')OR
14.  (K=' ▉')OR(K='▕')OR(K='▔')OR
15.  (K='▌')OR(K='▏')OR(K='▁')OR(K='▂')OR
16.  (K='円'%CR%)OR(K='－')OR
17.  (K='／'%KETU%)OR(K='」'%KETU%)OR
18.  (ORD(K)=102 %CR%)OR(ORD(K)=96 %DEL%)OR
19.  (ORD(K)=20 %■%)OR(K='月'%DEL%)OR
20.  (K='1')OR(K='2')OR(K='3')OR
21.  (K='4')OR(K='4')OR(K='5')OR
22.  (K='6')OR(K='7')OR(K='8')OR
23.  (K='9')OR(K='0');
24. WRITE(" ■");
25. KEYIN:=K;
26.END;
27.%-------------------------%
28.FUNCTION TENKEY:INTEGER;
29.VAR K:CHAR;
30.    N,IN:INTEGER;
31.    MINUS,KESSEKI:BOOLEAN;
32.BEGIN
33.  N:=0;MINUS:=FALSE;KESSEKI:=FALSE;
34.  K:=KEYIN;
35.  WHILE(K<>'円')AND(ORD(K)<>102)DO
36.   BEGIN
37.    IF(ORD(K)=96)OR(ORD(K)=20)OR(K='月')THEN
38.      BEGIN
39.        N:=N DIV 10;
40.        WRITE("■");
41.        IF ORD(CIN)=42 THEN MINUS:=FALSE;
42.                       MINUS:=FALSE;
43.      END
44.    ELSE IF K='－'THEN
45.      BEGIN
46.        MINUS:=TRUE;
47.        WRITE("－");
48.      END
49.    ELSE IF(K='／')OR(K='」')THEN
50.      BEGIN
51.        KESSEKI:=TRUE;
52.        WRITE("/");
53.      END
54.    ELSE
55.      BEGIN
56.        CASE K OF
57.          '0',' ▗':IN:=0;
58.          '1',' ▏':IN:=1;
59.          '2',' ▄':IN:=2;
60.          '3',' ▉':IN:=3;
61.          '4',' ▕':IN:=4;
62.          '5',' ▔':IN:=5;
63.          '6',' ▌':IN:=6;
64.          '7',' ▏':IN:=7;
65.          '8',' ▁':IN:=8;
66.          '9',' ▂':IN:=9;
67.          END;%CASE%
68.        WRITE(IN:1);
69.        N:=N*10+IN;
70.      END;
71.   K:=KEYIN;
72.   END;%WHILE%
73.  IF KESSEKI THEN N:=-1;
74.  IF MINUS THEN N:=-N;
75.  TENKEY:=N;
76.END;
77.%-------------------------%
78.BEGIN
79. WRITE("■");
80. REPEAT
81.   N:=TENKEY;WRITELN();
82.   WRITELN("N=",N:5);
83. UNTIL 3>5;
84.END.
85.%-------------------------%
86. %866 ｱﾘｼﾛｼ ｵﾓ ﾏﾁ 3922-12%
87.       %  ﾀﾞﾓﾄ ｺｳｽｹ %
88.
```

MZ-80K/C(Hu-BASIC)

パターン認識プログラム

……中田　勇

コンピュータに，自分の入力したパターンを判断させるプログラムを作りました。

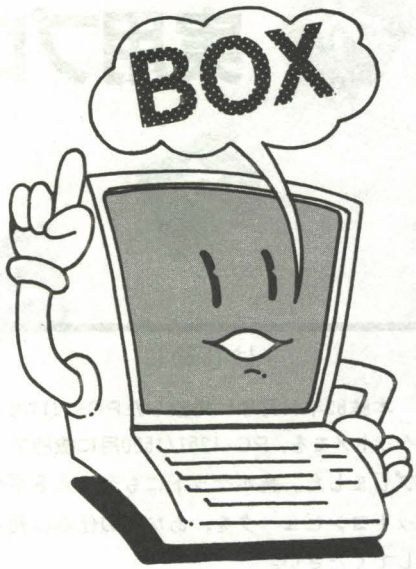
使い方は，キーボード上の複数のKEYを同時に押します。すると，コンピュータは次のような判断をしてくれます。

NONE	何もない
DOT	点である
BAR	棒のような形
BOX	箱のような形
TRI	三角のような形
RING	リングのような形
????	わからない

ただ，これだけのプログラムです。MZ-80K/Cのキー配列がキーボードマトリックスと同じなので一応成功しました。誤答率が高いと思われる方は改良してみてください。

◦プログラムおよび主な変数

10～60	マシン語
70～160	押したキーの数の計算
170～200	説明文
210～300	キースキャン
310～360	判別
K$(255)	押したキーの図
K(255)	押したキーの数
A	押したキーの２の列の数
B	押したキーの１の列の数
C	押したキーの０の列の数
D	押したキーの数の合計
P	押したキーの数の小計
Q	E001番地の値



リスト

```
10 '♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥
20  CLEAR &HCEFF
30  FOR J=0 TO 20:READ A$:POKE &HCF00+J,VAL("&H"+A$):NEXT
40  DATA 26,CF,2E,FF,7E,26,E0,2E,00,77
50  DATA 26,E0,2E,01,7E,26,CF,2E,FF,77,C9
60 '♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥♥
70 CLS:LOCATE17,10:PRINT"WAIT"
80 DIM K(255),K$(255),L$(1):X=255
90 L$(0)=".■":L$(1)="●■"
100 FOR A=0 TO 1:FOR B=0 TO 1
110 FOR C=0 TO 1:FOR D=0 TO 1
120 FOR E=0 TO 1:FOR F=0 TO 1
130 FOR G=0 TO 1:FOR H=0 TO 1
140 K$(X)=L$(H)+L$(G)+L$(F)+L$(E)+L$(D)+L$(C)+L$(B)+L$(A)
150 K(X)=A+B+C+D+C+D+E+F+G+H:X=X-1
151 PRINT X+1;K(X+1);K$(X+1)
160 NEXTH,G,F,E,D,C,B,A
170 CLS
180 LOCATE 17,5:PRINT"PATTERN ﾆﾝｼｷ"
190 LOCATE 17,8:PRINT"ﾌｸｽｳﾉ KEYﾗ ｻﾜｯﾃ ｸﾀﾞｻｲ"
200 LOCATE 17,10:PRINT"ｿﾉ PATTERNﾗ ｱﾃﾏｽ"
210 A=0:B=0:C=0:D=0:PRINT"■";
211 E=0:F=0
220 FOR I=0 TO 9 STEP 2
230 POKE &HCFFF,I:CALL &HCF00
240 Q=PEEK(&HCFFF):P=K(Q)
241 H=Q
250 PRINT K$(Q);"■■■■■■■■■■■■■■■■";
260 POKE &HCFFF,I+1:CALL &HCF00
270 Q=PEEK(&HCFFF):P=P+K(Q)
280 PRINT K$(Q);H;Q;"      "
290 A=A-(P=2):B=B-(P=1):C=C-(P>0):D=D+P
291 E=E-(P=3):F=F-(P=4)
300 NEXT
310 LOCATE 17,14
320 IF C=0 THEN PRINT"NONE":GOTO 210
330 IF D=1 THEN PRINT"DOT ":GOTO 210
340 IF A+B>3 OR (C=1 AND D>3) THEN PRINT"BAR ":GOTO 210
341 IF (B=1)+(A=1)+(E=1)+(F=1)<-1 THEN PRINT"TRI ":GOTO 210
342 IF D>8 AND A<1 THEN PRINT "RING":GOTO 210
350 IF D>3 THEN PRINT"BOX ":GOTO 210
360 PRINT"????":GOTO 210
```

注.) ３つ以上のキーを同時に押すと，押されていないキーまでも読みとることがありますが，ハード上しかたありません。

ポケコン横丁 実用プログラム「ポケットメモ」

Yuuji Sugiura & Kouji Kobayashi
杉浦勇二 & 小林光二

はじめに

本誌82年11月号に掲載したPC-1211用ポケットメモを，PC-1251/1500用に機能アップしました。胸ポケットにも楽々入るポケットコンピュータを，あなたの仕事に利用してください。

PC-1251/1500用ポケットメモの特徴

本プログラムは，データ格納エリアを増大し，キー部に1メモリ分(7バイト)，データ部に3メモリ分（21バイト）のデータを41組分格納することが可能です。データ入力時の操作も，I/Oバッファに79バイトの文字入力が可能，1回の入力でKEY部およびデータ部の入力を行ないます。

また，スケジューラ機能を追加し，IP(INPUT)でKEYに日付(MM/DD)を入力して，当日日付の入力内容を出力します。クリアすると，当日日付以前の入力内容をクリアします。このときプリンタ出力の指定をすると，消去した内容をすべてプリントアウトします。

そして次に，データのSAVEおよびLOAD処理を正確に行なうために，カセット・インタフェイスを追加しました。本サブルーチンは，CE-125プリンタのLPインタフェイスとして，PRINT命令をLPRINT命令に置き換えています。

操作方法

PC-1500用はまず最初に，NEW & 4100 [ENTER] と必ず行なってください。PC-1500では4K RAM CF-151が必要です。

プログラムを正しく入力したらCLEAR [ENTER] とし，[DEF] [S] によって起動すると，?LP-OUT Y/N=と表示が出ます。このとき6行でエラーが出るようでしたら，RUNで起動してください。

CE-125またはCE-150のプリンタが接続してある場合は，Yを入力するとプリンタに出力されます。通常はNを入力するか，[ENTER] キーを押します。次にメニューが表示されます。

F/IP/DL/TR/SC=

2度目以降の起動は [DEF] [S] です。

A) IP（INPUT）指定

データを入力するときのコマンドです。[ENTER] キーだけを押すとメニューに戻ります。1500用は [!] で戻ります。また，検索キー名だけ入力してもエラーとして扱われ，記憶されません。検索キーは，6文字以上，PC-1500用では15文字以上は無視します。デルミタは〝:〟とし，入力データは入力可能なだけ，I/Oバッファは79文字までOKです。

```
表示  ?
入力  TOKYO : JAPAN␣WORLD (ENTER)
      検索キー  ↑デルミタ  入力データ内容
```

B) F（FIND）指定

検索キーを入力することで特定のデータを探し出し，表示します。

```
表示      ?
入力      TOKYO (ENTER)
処理結果  JAPAN␣WORLD
```

C) DL（DELETE）指定

記憶されているデータの一部を削除，同一の検索キー名のデータはすべて消えます。

D) TR（TRACE）指定

データをすべて表示していきます。

E) SC（SCHEDULE）指定

検索キーを月/日として入力しておき，スケジュールとして使います。当日の日付のデータがある場合は内容を，ない場合はメニューを表示します。クリア指定をした場合は当日日付以前のデータはDELETEされます。

```
表示  ? CLEAR Y/N= _
入力  Y
表示  ?
入力  01/25 (ENTER)←1月25日の意味です。
```

カセット・インタフェイスの起動法

[DEF] [C] で起動させると，1)SAVE 2)LOAD= と表示されます。データのセーブでは1を押すと，RDY? MEN>> CMTと表示され，録音の用意ができたら [ENTER] を押します。次にコメント，カセット番号(0 or 1)，テープの開始カウントを入力すると，データの出力が始まります。終了するとプリンタに印字されます。データをロードする場合は [DEF] [C] のあと，2を押してください。

おわりに

今回はPC-1211の移植版の「ポケットメモ」を，さらにPC-1500でも使えるようにしました。アルゴリズムは多少異なりますが，同等のところは同じ行番にしてあります。

PC-1251では，データの記憶は41組までですが，PC-1500では16字以下のデータなら100組まで可能です。

今回使ったマシン語部分は，PRINTとLPRINTを変換させるものです。PC-1250にはあるこの命令が，PC-1500では見当たりませんので，マシン語で作りました。機会がありましたら，PC-1500でのマシン語の使い方について説明したいと思います。

とにかく，今回のポケットメモ，なかなか便利です。アドレス帳やスケジュール表として有効に使ってください。

リスト（PC-1251用）

```
 5:REM *MEMO* V/R=2/3
 6:"S" IF I=0 LET I=1:
   DIM B$(2)*80
 8:GOSUB "LPIF"
10:GOTO 100
11:"C"X=0:Y=0:Z=0:I=0
12:BEEP 1: INPUT "1)SAV
   E 2)LOAD =";Z$
13:IF Z$="1" GOTO 20
15:PRINT "RDY? MEM<<CMT
   "
16:CLEAR : INPUT #"DATA
   "
17:I=0
18:LPRINT "LOAD OK COME
   NT=";C$: LPRINT ""
19:END
20:PRINT "RDY? MEM>>CMT
   "
22:BEEP 1: INPUT "COMEN
   T=";C$
23:BEEP 1: INPUT "C-MT
   NO=";I
24:BEEP 1: INPUT "START
   -COUNTER=";J
26:Z$=" DATA  ": PRINT
   #"DATA"
30:BEEP 1: INPUT "END-C
   OUNTER=";M
31:LPRINT "": LPRINT ""
32:X$="-------"
33:LPRINT X$;Z$;X$
34:LPRINT "COMENT=";C$
35:Y$="#"+ STR$ I:Z$=
   STR$ J+"-"+ STR$ M
36:LPRINT "  C-MT= ";Y$
   ;" (";Z$;")"
37:LPRINT X$;X$;X$
38:LPRINT ""
39:I=1: END
100:D=27
105:I=41:J=4: REM 41-ENT
    RY
110:A(D)=I:A(D+1)=J
120:GOSUB "DIM":A(A+1)=0
130:"A"K$="": BEEP 1:
    INPUT "F/IP/DL/TR/SC
    =";Z$
135:GOTO Z$
250:"IP"J=1:I=1: GOSUB "
    IKD"
255:IF I>A(D) BEEP 5:
    PRINT "AREA NOTHING"
    : GOTO "A"
260:GOSUB "DIM"
265:IF A$(A) LET I=I+1:
    GOTO 255
275:A$(A)=K$:A$(A+1)=E$:
    A$(A+2)=F$:A$(A+3)=G
    $
276:L= LEN B$(2)
277:IF L<=21 GOTO "IP"
278:B$(2)= MID$ (B$(2),2
    2,L)
279:GOSUB 606
280:I=I+1
282:GOTO 255
300:"F" GOSUB "IKD"
303:J=1:I=1
304:WAIT 1: PRINT K$;" :
    ": WAIT
305:IF I>A(D) GOTO "A"
310:GOSUB "DIM"
315:IF A$(A)<>K$ LET I=I
    +1: GOTO 305
320:WAIT : GOSUB "PRT"
322:IF LEN B$(0)-3<21
    WAIT 1: PRINT ""
325:I=I+1: GOTO 305
400:"DL" GOSUB "IKD"
405:J=1:I=1
410:IF I>A(D) GOTO "A"
415:GOSUB "DIM"
420:IF A$(A)<>K$ LET I=I
    +1: GOTO 410
425:A(A)=0
430:I=I+1: GOTO 410
450:"TR"J=1:G$=""
460:FOR I=1 TO A(D)
465:GOSUB "DIM"
468:IF A$(A)=G$ GOTO 475
470:PAUSE A$(A);" : "
471:B$(0)=A$(A+1)+A$(A+2
    )+A$(A+3)
472:WAIT 1: PRINT A$(A);
    " :"
473:PRINT "    ";B$(0)
474:WAIT
475:NEXT I
480:GOTO "A"
499:BEEP 1: PAUSE "ERR !
    "
500:"SC"Z$="N": BEEP 1:
    INPUT "CLEAR? Y/N ="
    ;Z$
501:BEEP 1: INPUT "DATE?
    MM/DD =";U$
502:IF LEN U$<>5 GOTO 49
    9
503:IF U$>"12/32" GOTO 4
    99
508:K$=U$
509:IF Z$<>"Y" GOTO 303
510:J=1:G$=""
515:FOR I=1 TO A(D)
516:GOSUB "DIM"
519:IF A$(A)=G$ GOTO 530
520:IF A$(A)<U$ WAIT 1:
    PRINT "<DEL>";A$(A):
    WAIT 60: GOSUB "PRT"
    :A(A)=0
530:NEXT I
535:GOTO 303
540:GOTO "A"
600:"IKD" PAUSE "KEY:":
    PAUSE "MM/DD:"
601:BEEP 1: INPUT B$(0)
602:GOSUB "FECH"
604:IF C=1 LET K$=B$(1)
606:K$=B$(1):E$=B$(2):F$
    = MID$ (B$(2),8,7):G
    $= MID$ (B$(2),15,7)
608:RETURN
620:"PRT"B$(0)=""
622:B$(0)="   "+A$(A+1)+
    A$(A+2)+A$(A+3)
624:BEEP 1: PRINT B$(0)
630:RETURN
800:"FECH" REM FECH
802:C=0:N=1
804:IF N> LEN B$(0) GOTO
    820
806:Z$= MID$ (B$(0),N,1)
808:IF Z$<>":" LET N=N+1
    : GOTO 804
810:C=C+1:L=N-1
812:B$(C)= MID$ (B$(0),1
    ,L)
813:N=N+1
814:L= LEN B$(0)-N+1
815:B$(0)= MID$ (B$(0),N
    ,L)
816:N=1: GOTO 804
820:L= LEN B$(0)+1:C=C+1
821:IF C>2 GOTO 824
822:B$(C)= MID$ (B$(0),1
    ,L)
824:RETURN
830:"LPIF" BEEP 1: INPUT
    "? LP-OUT Y/N=";Z$
831:IF Z$="Y" PRINT =
    LPRINT
832:IF Z$="N" PRINT =
    PRINT
833:RETURN
900:"DIM"A=A(D+1)*(I-1)+
    J+D+1: RETURN


3486- MEM
                 , 3457.
```

リスト（PC-1500用）

```
 5:REM  *MEMO* PC
   -1500
 6:"S"IF I=0LET M
   =200:DIM B$(1)
   *80,A$(M):I=1:
   A$(0)=""
 8:GOSUB "LPIF"
10:GOTO 130
11:"C"X=0:Y=0:Z=0
   :I=0:WAIT
12:BEEP 1:INPUT "
   1)SAVE 2)LOAD
   =";Z$
13:IF Z$="1"GOTO
   20
15:PRINT "RDY? ME
   M<<CMT"
16:INPUT #"DATA";
   A$(*),C$
17:LPRINT "LOAD O
   K"
18:LPRINT "COMENT
   =";C$:LPRINT "
   "
19:END
20:PRINT "RDY? ME
   M>>CMT"
22:BEEP 1:INPUT "
   COMENT=";C$
23:BEEP 1:INPUT "
   C-MT NO=";I
24:BEEP 1:INPUT "
   START-COUNTER=
   ";J
26:Z$=" DATA  ":
   PRINT #"DATA";
   A$(*),C$
30:BEEP 1:INPUT "
   END-COUNTER=";
   E
31:LPRINT "":
   LPRINT ""
32:X$="--------"
33:LPRINT X$;Z$;X
   $
34:LPRINT "COMENT
   =";C$
35:Y$="#"+STR$ I:
   W$=STR$ J+"-"+
   STR$ E
36:LPRINT "C-MT="
   ;Y$;" (";W$;")
   "
37:LPRINT X$;X$;X
   $
38:LPRINT ""
39:I=1:END
130:"A"K$="":BEEP
   1:INPUT "F/IP/
   DL/TR/SC=";Z$
132:IF Z$="F"OR Z$
   ="IP"OR Z$="DL
   "OR Z$="TR"OR
   Z$="SC"GOTO Z$
134:GOTO 130
250:"IP"A=0:GOSUB
   "IKD":IF B$(0)
   ="!"GOTO "A"
251:IF K$=B$(0)
   BEEP 1:PAUSE "
   ERR!":GOTO "IP
   "
255:IF A>MBEEP 5:
   PRINT "AREA NO
   THING":GOTO "A
   "
265:IF A$(A)LET A=
   A+1:GOTO 255
273:A$(A)=MID$ (K$
   ,1,14)+" :"
274:A=A+1:IF A>M
   GOTO 255
275:A$(A)=B$(0)
276:L=LEN B$(0)
277:IF L<=16LET A$
   (A+1)="":GOTO
   "IP"
278:B$(0)=MID$ (B$
   (0),17,L-16)
282:GOTO 274
300:"F"GOSUB "IKD"
   :IF B$(0)="!"
   GOTO "A"
303:A=0
305:IF A>MOR A$(A)
   =""GOTO "A"
315:IF A$(A)<>K$+"
   :"LET A=A+1:
   GOTO 305
320:WAIT 20:PRINT
   A$(A):WAIT
321:A=A+1:IF A$(A)
   <>""AND RIGHT$
   (A$(A),2)<>" :
   "PRINT "  ";A$
   (A):GOTO 321
322:WAIT 1:PRINT "
   "
325:GOTO 305
400:"DL"GOSUB "IKD
   ":IF B$(0)="!"
   GOTO "A"
405:A=0
410:IF A>MOR A$(A)
   =""GOTO "A"
415:IF A$(A)<>K$+"
   :"LET A=A+1:
   GOTO 410
420:GOSUB 421:GOTO
   410
421:J=A:A=A+1
422:IF A>MLET A$(J
   )="":RETURN
423:IF RIGHT$ (A$(
   A),2)<>" :"LET
   A=A+1:GOTO 422
424:FOR P=JTO M
425:A$(P)=A$(P+A-J
   ):IF A$(P)=""
   RETURN
426:NEXT P
430:RETURN
450:"TR"
460:FOR A=0TO M
468:IF A$(A)=""
   GOTO "A"
470:IF RIGHT$ (A$(
   A),2)=" :"GOTO
   474
472:WAIT 20:PRINT
   "  ";A$(A):
   GOTO 475
474:WAIT 60:PRINT
   A$(A)
475:NEXT A
480:GOTO "A"
499:BEEP 1:PAUSE "
   ERR !"
500:"SC"Z$="N":
   BEEP 1:INPUT "
   CLEAR? Y/N =";
   Z$
501:BEEP 1:INPUT "
   DATE? MM/DD ="
   ;U$
502:IF LEN U$<>5
   GOTO 499
503:IF U$>"12/32"
   GOTO 499
508:K$=U$
509:IF Z$<>"Y"GOTO
   303
515:FOR A=0TO M:Q=
   A
516:IF RIGHT$ (A$(
   A),2)<>" :"
   GOTO 530
519:IF A$(A)=""
   GOTO 530
520:IF A$(A)<U$+"
   :"WAIT 20:
   PRINT "<DEL>";
   A$(A):WAIT 40:
   GOSUB "PRT":
   GOSUB 421:A=Q:
   GOTO 516
530:NEXT A
535:GOTO 303
600:"IKD"PAUSE "KE
   Y:":PAUSE "MM/
   DD:"
601:BEEP 1:B$(0)="
   ":INPUT B$(0):
   IF B$(0)="!"
   RETURN
603:N=1:IF B$(0)="
   "GOTO 601
604:IF N>LEN B$(0)
   LET K$=B$(0):
   GOTO 619
606:Z$=MID$ (B$(0)
   ,N,1)
608:IF Z$<>":"LET
   N=N+1:GOTO 604
610:L=N-1
612:K$=MID$ (B$(0)
   ,1,L)
613:N=N+1
614:L=LEN B$(0)-N+
   1
615:B$(0)=MID$ (B$
   (0),N,L)
619:RETURN
620:"PRT"IF A+1>M
   RETURN
621:FOR J=A+1TO M
622:IF A$(J)=""OR
   RIGHT$ (A$(J),
   2)=" :"RETURN
624:BEEP 1:PRINT "
   ";A$(J)
625:NEXT J
630:RETURN
830:"LPIF"BEEP 1:Z
   $="N":INPUT "?
   LP-OUT Y/N=";
   Z$
831:IF Z$="Y"GOSUB
   "LP"
832:IF Z$="N"GOSUB
   "P"
833:RETURN
900:"LP"Y=&B9:X=&9
   7:GOTO 920
910:"P"X=&B9:Y=&97
920:POKE &40E0,&48
   ,&44,&4A,&40,&
   44,&FD,&5A,&5C
930:POKE &40E8,&4A
   ,&8B,&10,&B5,&
   F0,&07,&99,&0C
940:POKE &40F0,&44
   ,&B5,X,&07,&99
   ,&12,&B5,Y
950:POKE &40F8,&0E
   ,&9E,&17,&9A
960:CALL &40E0:
   RETURN
```

Oh!MZ 質・問・箱

Q MZ-700を使っているのですが，S・Hu-BASICのデモプログラムで中間色が出ているようです。もしあれが中間色ならば，その出し方を教えてください。

静岡県　宮津義之

A 同じような質問が，広島県の森下由隆さん，神奈川県の佐藤仁さん，ほか何人かの人から寄せられています。

S-BASIC，Hu-BASIC，どちらのテープの裏側にもAPPLICATIONプログラムが入っています。これを走らせるとたしかに茶色やピンクの中間色が使われています。

中間色を出すにはドット単位に色を交互に変えていけばよいのです。MZ-700はグラフィックが使えないので，かわりにキャラクタの ▒ を使います。このキャラクタは GRAPH キーを押して Z を押すと出てきます。中間色を出す簡単な例として，下のリストをHu-BASICで走らせてください。フォアグラウンド・カラー（キャラクタ自体につく色）とバックグラウンド・カラー（背景につく色）を0から7の範囲でいろいろと変えてみましょう。

COLOR2,7では茶色（はだ色に近い），COLOR2,3でピンクのように見えます。

ただし，四角形のキャラクタ単位でしか使えないため，使用範囲は限られてきます。

（杉浦勇一）

```
10 CLS:Y=1
20 LOCATE 0,0:COLOR 7,0
30 INPUT "FOR. COLOR (0-7)";F
40 INPUT "BACK COLOR (0-7)";B
50 COLOR F,B
60 FOR I=1 TO 5
70   LOCATE X,Y+I
80   PRINT "▒▒▒▒▒▒▒▒"
90 NEXT I
100 X=X+5:IF X> 39 THEN X=0:Y=Y+5
110 Y=Y MOD 20
120 GOTO 20
```

Q X1について質問します。X1のHu-BASICにはMODという命令があるようですが，BASICのマニュアルには載っていません。この命令の意味を教えてください。

例．X1テクニカルマスター117ページ（プログラム4－16）

110 IF （I-1）MOD 8=0 THEN～

岡山県　林　亮

A MODについてはBASIC MANUALの11ページに少しだけ説明があります。しかし，索引にも出ていないので「載っていない」と思われたのでしょう。

MODというのは命令（コマンド及びステートメント）の部類ではなく，演算子に入ります。つまり，＋－＊／と同じ仲間なのです。その意味は整数剰余，つまり，「割ったときのあまり」を求めるものです。

上の例では，I－1を8で割ったときのあまりが0のとき～の処理を行なうという意味です。このMODを使う人はたいへん少ないようなので，実際にどのように使うかの1例をあげておきます。

前の質問で使ったリストを見てください。行30と40色で色の値を0から7と指定して入力を要求しています。もし，間違えて8を入れた場合50行でエラーとなってしまいます。色の値は7までしかないからです。

ここで50行を

COLOR F MOD8, B MOD8

と変えたとします。「F MOD 8」というのは変数Fを8で割ったあまりでしたね。これなら，変数Fにどんな大きな数値を入れても8以上になることはありません。よって，7を入力したときは7÷8=0…7で7のままですが，8を入力したときは，8÷8=1…0と，0に戻されてしまいます。

同じ理由で110行でも使われています。たいていの人はこのような場合

110 IF Y>20 THEN Y=1

としますが，MODを使ったほうが短くてすみます。

（杉浦勇一）

Q MZ-731のユーザーです。Oh!MZ 3月号，11月号の「IPL27」，「カセットオペレーティング」で早くローディングできるようになり，便利に使っています。しかし，ヘッドのちょっとした汚れでローディングできなくなります。そこで，2700ボーではなくて2000ボーぐらいに変更できないでしょうか。

広島県　永野常雄

A 要は，「ボーレートの変更」の部分をタイミングがずれないように，少し大きな値に変えれば良いのです。

Hu-BASICの場合で説明します。まず，1983年11月号の「CMT/OS」を入れ，走らせます。そして，11月号42ページの説明どおりにHu-BASICをロードし，モニタのMコマンドで次のように変更します。

$2021 $6F→$3C　$202E $6F→$3C
$2040 $36→$1E　$204A $34→$1C
$2178 $4F→$2C

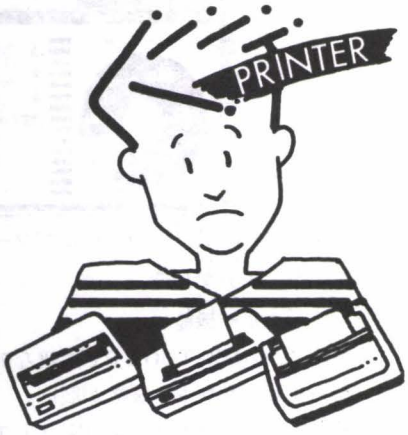
これでボーレートは2000ボーに近い値になります。ただし，Hu-BASICでこのままではMZ-2000/2200用のプログラムを読むことはできません。

2700ボーのときよりもセーブ・ロードの際の信頼性は増すはずです。（大西　太）

Q MZ-2200を持っています。先ごろシャープから漢字カラーBASICとともに漢字ROMボードが発売されましたが，このボードはI・Oデータ機器の漢字ROMボードと互換性はありますか。

石川県　安川俊明

A MZ-2000の発売当初からI・Oデータ機器より漢字ROMボード（PIO-3055，35,000円）が発売されていましたが，今度シャープが発売したMZ-1R13はこのPIO-3055にオプションのかな漢字変換ROM（PIO-3055-01，7,000円）を実装した



ものです。基板は両方とも同一ですが，ポートアドレス設定のディップスイッチがI・Oデータ機器製ではありましたが，シャープ製はジャンパー線により固定されています。このディップスイッチの省略もあってか，シャープ製のほうが200円ほど安いようですが，基本的には同じ物といえます。したがってどちらを使用しても，漢字カラーBASIC(MZ-2Z021)やHuBASIC Ver 2.0が何ら支障なく使えますので，問題はありません。　（松崎　洋）

Q　MZ-5521を持っています。近くプリンタを購入しようと思いますが，シャープの広告によると5種類（MZ-1P02，1P03，1P04，1P06，1P07）あり，どれにしようか迷っています。選ぶときのアドバイスをお願いします。　大阪府　吉田一義

A　パーソナルコンピュータはシステムを構築してはじめて本来の性能を発揮します。本体だけあってもCRTがなければ使えませんし，ビジネスに利用する場合，帳票を出力するプリンタがなければだめですね。せっかく優れたパソコンを手に入れたのですから，システムを上手に活用するには機種選択は重要です。もちろん予算にもよりますが，目的に合ったものを選ぶべきでしょう。

MZ-5500シリーズ用プリンタは，ご質問にもありましたとおり，現在5タイプが用意されており，

MZ-1P02　8×8ドット，80桁
MZ-1P03　24×24ドット，136桁，漢字
MZ-1P04　カラーインクジェット
MZ-1P06　16×16ドット，80桁，漢字
MZ-1P07　8×8ドット，80桁

となっています。当然のことながらドットが多くなれば繊細な文字が印字でき，単に性能の面ではMZ-1P03がベストかと思いますが価格が54万円と高価で，誰でも入手できるというものではありません。

この5種のうち，新たにMZ-5500シリーズ用として発売されたものはMZ-1P06のみで，MZ-1P02，-1P03はMZ-3500シリーズ，MZ-1P07はMZ-2200ですでに発売されていたものがMZ-5500シリーズで使えるといったことにすぎません。したがってインタフェイス自体セントロニクス準拠のパラレルのプリンタであれば，エプソンのFP-80や日本電気のPC-PR201なども使えるのです（ただしケーブルは，シャープ純正のMZ-1C32を使います）。

シャープ5種の純正プリンタに限って考えてみますと，CP/M-86などOS上では一応すべてサポートされています。しかし，MZ-1P02や1P07では漢字を含んだリスト出力ができませんし，MZ-1P04のカラーハードコピーも漢字CP/M-86でしかサポートされていません。またキャラクタフォントも一部異なりますので，仮にアルファベットの反転文字などMZ-5500シリーズ特有の文字を含んだリストをプリントアウトした際いわゆる化け字になってしまい，若干の不都合が生じるのです。これは他社のセントロニクスI／Fのプリンタに関しても同様です。ただし，ダウンロードができる機種はこの限りではありません。

基本的な件は以上のとおりです。さて，けっこう見落しがちなことですが，市販のパッケージソフトを使用する場合にそのソフトがどのプリンタをサポート（使用できるか否か）しているかも考えておかなければなりません。単なる表集計形式のデータベースや簡易言語であれば特に問題が生じることもないのですが，日本語処理つまり日本語ワードプロセッサともなりますと事態は極めて深刻です。8ピンのプリンタを使用してのワープロは2パス印字（千鳥で1パス印字のあと1／216インチずらしてさらにその上に印字する）方式がもっともよく使われ，16ピン24ピンでは1パスで印字しています。いずれもキャラクタコードをプリンタに送りそれを印字する方式とは異なり，ビットイメージやペーパーフィード等プリンタを細かく制御しなくてはなりませんので，原則として指定外プリンタは使用できないのです。

もし，吉田さんがワープロ利用を主目的としてプリンタを選ばれるのでしたら，MZ-1P03かMZ-1P06をお推めします。MZ-1P03は24ピンプリンタで漢字を出力した場合明朝体に近い高品質の出力が得られます。MZ-1P06は16ピンのためどちらかといえばゴシック体に近い印字ですが実用には十分でしょう。なお用紙は10インチ幅，つまり単票にしてA4サイズ縦までが限界です（MZ-1P03は16インチです）。

シャープはあまり周辺機器の広告を出していないようなので，仕様等細かくチェックができませんが，同じ純正とはいってもMZ-5500シリーズ用で市販されたものは今のところMZ-1P06だけですからワープロをあまり考えていない場合でもこのプリンタを選ばれた方が後々無難かと思います。なお各種プリンタの詳しい仕様はメーカーに問い合わせてください。　（青木秀浩）

質問にお答えします

日ごろ疑問に思っていること，どんなことでも結構です。どんどんお便りください。難問，奇問，編集部が総力をあげてお答えいたします。ただし，お寄せいただいているものの中には，マニュアルを読めばすぐに解答が得られるようなものも多々あります。最低限，マニュアルは熟読しておきましょう。質問はなるべく具体的に，機種名，システム構成，必要なら図も入れてこと細かに書いてください。また，返信用切手同封の質問をよく受けますが，原則として，質問には本誌上でお答えすることになっていますのでご了承ください。なお，質問の内容について直接問い合わせることもありますので，電話番号も明記してくださいね。

宛先：〒102 東京都千代田区四番町2－1
(株)日本ソフトバンク
出版部「Oh！MZ質問箱」係

マルチウィンドウ・マウスで簡単操作

MZ-5500に話題の統合化ソフト"TODAY"登場!!

シャープは，MZ-5500の統合化ソフトである“TODAY”を発売しました。

TODAYは，MZ-5500の特長であるマルチウィンドウ，マウス，ビットマップディスプレイなどの機能をフルに引き出すソフトで，MZ-TOOL（統合化のための管理プログラム）を核として，その上にMZ-CALC(表処理計算用)，MZ-GRAPH(グラフ処理用）を構成し，データの一元化を可能にしています。

このTODAYにより，あたかも頭の中にいくつかの局面を描いて作業を進めるという人間の基本的な思考プロセスに近い形での実行が可能です。

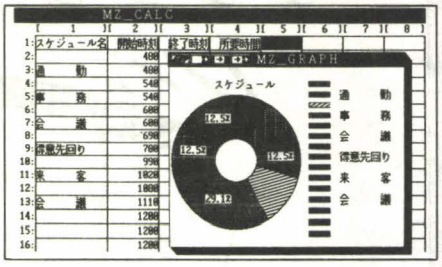
〈TODAYの特長〉

1)マルチウィンドウ機能により，複数のアプリケーションの実行を任意に切り換えながら，CRT上に自由なレイアウトで複数の作業を同時に表示できる。

2)マウスによる操作および，漢字を使用したガイダンスと豊富なヘルプメッセージによる簡単な操作が可能。

3)各アプリケーションで扱うデータを共通管理，複数のアプリケーション内での利用が可能。

4)他のアプリケーションとデータ交換可能。



5)辞書ROM（MZ-1R14)の使用でワープロ並の漢字入力が可能。

6)ハードコピー機能。

7)今後，このMZ-TOOL上で広汎なアプリケーションソフトウェアが展開。

8)TOOL上のアプリケーションとして今回MZ-CALCとMZ-GRAPHを統合化。

MZ-2Z014 5D3枚組 68,000円

ポケットコンピュータ

PC-1260／PC-1261を新発売

シャープでは，BASICを知らなくても買ったその日から使いこなせるビジネスシミュレーションソフトを搭載した手帳サイズのポケットコンピュータPC-1260／PC-1261を新発売します。また，本体とデザインを統一したプリンタとマイクロカセット一体化の周辺機器CE-125Sも同時発売します。

〈PC-1260／1261の特長〉

1)計算式をそのままの形で入力できるビジネスシミュレーションを搭載。

2)手帳サイズで24桁×2段表示

3)プログラム作成に便利なヘルプ(HELP)機能を採用。

4)カタカナ表示機能により，わかりやすい実用的なプログラムが可能。

5)大きなメモリ容量。RAM4.4Kバイト(PC-1260)，10.4Kバイト（PC-1261）

6)PC-1250のBASICをベースにして，一段とレベルアップした高度なBASIC。またパスワード（暗証番号）機能付き。

7)操作性を配慮した使い易い設計。また，ラストアンサー機能により，直前に行なった計算の結果を効果的に使用できる。

価格は，PC-1260が29,800円，PC-1261が39,800円，CE-125Sが36,800円。

乳牛の健康にもMZ!?

山形県にある大蔵ホルスタイン同志会では，乳牛の健康と繁殖のためにMZ-2000を導入し，合理的な飼料計算を行なっています。乳牛の能力は乳量と繁殖で，そのバランスが大切です。酪農家にとって，体重や体質に応じて適切な飼料を与えることが重要ですが，この計算を電卓ですると一頭で30分以上もかかります。同会ではMZを利用して計算時間を2～3分に短縮，さらに6ヶ月間の平均飼料も約30万円削減出来たとのことです。

使用したソフトは，同山形県天童市の酪農家が開発したもので，土地に合わせてアレンジしているそうです。①飼料計算の他に②乳量推定もあり，日本マイコンセンター山形にて取り扱っています。

飼料計算システムMZ-2200／2000／80B-T-30,000円　日本マイコンセンター山形　☎0236－41－0223

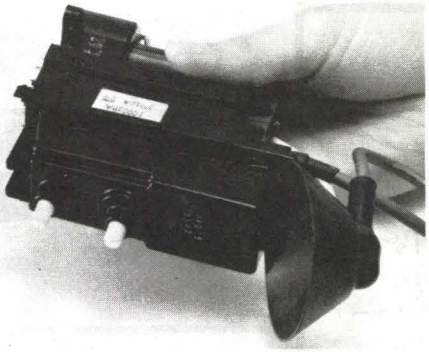
バックナンバー常備店

（東京）		
神保町	三省堂神田本店5F	03(233) 3312
〃	書泉ブックマートB1	03(294) 0011
八重洲	八重洲ブックセンター3F	03(281) 1811
渋谷	東急ハンズ寿楽洞7F	03(464) 4604
新宿	紀伊国屋書店本店	03(354) 0131
〃	未来堂書店	03(200) 9185
池袋	西武百貨店マイコン売場9F	03(981) 0111
調布	真光書店	0424(87)2222
町田	東急ハンズ寿楽洞	0427(28)2782
水戸	川又書店駅前店	0292(31)0102
柏	新星堂カルチェ5	0471(64)8551
川越	黒田書店	0492(25)3138
横浜	有隣堂ルミネ店	045(453)0811
船橋	西武ブックセンター10F	0474(25)0111
愛知・豊橋	精文館書店	0532(54)2345
京都・中京区	オーム社書店	075(221)0280
大阪・都島区	駸々堂京橋店	06(353) 2413
藤沢	有隣堂藤沢店	0466(26)1411
津田沼	芳林堂書店パルコ5F	0474(78)3737
長野・飯田	平安堂飯田店	0265(24)4545
川口	岩渕書店	0482(52)2190
厚木	有隣堂厚木店	0462(23)4111

シャープ――小型・高性能設計　カラーディスプレイ用フライバックトランスを開発・発売

シャープでは，小型・高性能設計のカラーディスプレイ用フライバックトランス，〈F0008〉を開発し，新発売します。

ここ数年CRTディスプレイ市場は活況を呈しており，特にカラーCRTディスプレイは著しい成長を続けています。さらに，OAニューメディアの進展に伴いCRTディスプレイは高品位化の傾向にあり，搭載されるフライバックトランスに対し，高性化の要求が強まっています。



このたび発売するフライバックトランス〈F0008〉はこうした市場の要望にお応えするもので，省スペース指向に徹した小型設計に加え，高圧コンデンサを内蔵，トッププロード方式を採用し，高圧変動率の低減およびフォーカス電圧の無変動を実現した高性能・高信頼度設計になっています。

4月1日発売　4,700円

YDK――春の入学・就職お祝セール

マイコンショップ「YDKシステムセンター」では，3月20日から4月30日まで，入学・就職お祝いセールを実施中です。

期間中サプライ品，MSXパソコン，その他パソコンをシステムでお買い上げの方に♥ディスケットケース　♥カシオデータバンク　♥腕コン　など，もらってうれしいわくわくプレゼントを差上げます。

（新宿NSビル1 F☎03-342-2441）

パソコンサンデー 第5期スタート

去る2月に通算100回を迎えたパソコン情報番組「パソコンサンデー」は，4月より第5期がスタートします。

第5期は，Dr. パソコン宮永先生を講師に，司会は下条アトムと坂上味和が担当。MZ-5500を使ったBASIC講座をはじめ，女子大生のパソコンレポート，パソコンソフトの紹介など，今までのシリーズ以上にパソコン情報色の強い番組作りをしていきます。

テレビ大阪・テレビ東京・テレビ愛知など全国21局にて4月1日より放映開始。

毎週日曜日　9：30～10：00

〈番組内容〉

1)パソコンBASIC講座
2)パソコン生活最新情報
3)今週のBESTパソコンソフト
4)パソコンサイエンス
5)パソコンワンポイントクエスチョン
6)パソコングラフィクス

〈テキスト〉

「パソコンサンデー第5期テキスト」
新紀元社刊　980円

第2回　月間特別賞 工藤恵氏「インタプリタパスカルの拡張」に決定

Oh！MZ月間特別賞，2月末日締め切り分は，「インタプリタパスカルの拡張」の工藤恵氏に決定しました。BASICの次の言語としては最適といわれるPASCALですが，残念ながらDISK版は発売されていませんでした。DISKを所有されている方々には，PASCALがより身近かになったことと思われます。工藤さんには，記念品として，ソニーWALKMAN（WM-20）をお送りいたします。

マイクロコンピュータショウ'84 ――高度情報化社会とマイコンの役割――

マイクロコンピュータショウ（日本電子工業振興協会主催）が，今年も来たる5月23日(水)～26日(土)の4日間，東京流通センター大展示場（平和島）にて開催される。

マイコンショウは，マイコンファンにとって最大のイベントで，昨年は114社が参加，なんと10万人もの観客を動員した。今年もすでに114社の参加が決定しており，各社とも，それぞれに趣向をこらした演出で，マイコンの新しい可能性を見せてくれるに違いない。また，例年同ショウにあわせて新製品を発表するメーカーも多いので見逃すことができない。期待のシャープのブースは，第1会場にある。受付から入ると右手奥の1―14なので迷わず直行しよう。

○第2会場2―42には日本ソフトバンクのブースもあり，出展にそなえて準備中です。ぜひ立ち寄って見てください。

愛読者プレゼント

1 キャリーラボ ☎0963-63-0211

スターライト・アドベンチャーⅠ，Ⅱ

A. X1用
B. MZ-2200/2000用　**各10名**

画面ひとつのテキストアドベンチャー型のPARTⅠときれいなグラフィックのPARTⅡを2本セットでプレゼントします。アドベンチャーはコマンド入力が面倒だ，という方も短縮入力でラクラク。しかし，謎は難しいですよ。各機種PartⅠ，3,500円　PartⅡ，3,800円

2 BOND SOFT ☎0534-53-6186

タイムシークレット

A. X1用
B. MZ-80K/C用　**各10名**
C. MZ-700用

このパッケージはMZ-700用です。

本誌3月号でも掲載し，好評を得ているアドベンチャーゲーム「タイムシークレット」に，X1用，MZ-80K/C用ができました。MZ-700ユーザーのみの楽しみ独占と嘆いていたキミ，応募しようよ。

各機種3,300円

3 BASIC HOUSE ☎0286-33-1994

AUTOCRACY

MZ-700用
10名

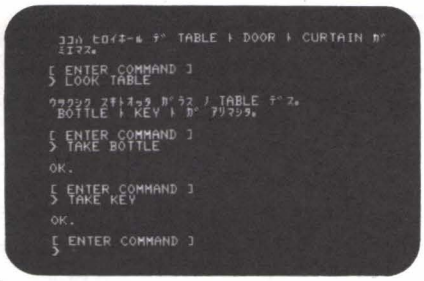
アップル風本格的アドベンチャーゲームと銘打つ，画面数120枚の超大作です。オール英語のゲームですから，英語の勉強にも役立つかも……。今号の「ADVENTURE，アドベンチャー」に紹介があります。　3,800円

4 マイクロキャビン ☎0593-51-6482

不思議の国のアリス

MZ-2200/2000/80B用　**10名**

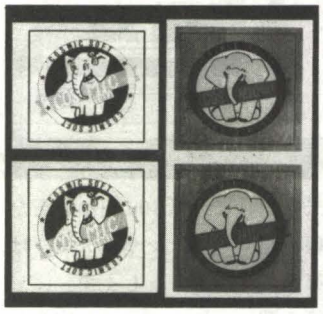
今号でアドベンチャーゲーム作りのテクニックを公開してくれたマイクロキャビンより，テキストとなった「不思議の国のアリス」を提供していただきました。想像力を働かせながら不思議の国へどうぞ。　3,500円



5 ダイツー ☎06-344-5150

COSMIC SOFTシール

2セットを**50名**



「デジタルペインター」「アニメディター」などのお絵描きツールを開発している(株)ダイツーより，かわいい象のキャラクタのCOSMIC SOFTシールをプレゼントしていただきました。アンチックな雰囲気のシールです。

プレゼントの応募方法

とじ込みのアンケートはがきの該当項目をすべてご記入のうえ，希望のプレゼント番号をはがき右上のスペースにひとつ記入してお申し込みください。締め切りは5月15日の到着分までとします。当選発表は7月号で行ないます。

3月号プレゼント当選者発表

①Ⓐ (茨城県)飯村はるみ　(愛知県)栗原周作　(奈良県)中西英樹　(広島県)大野保則　(大阪府)阿部賢一　Ⓑ (東京都)橋岡正和　(石川県)川畑健　(兵庫県)村山幸広　(長野県)牛山善行　(東京都)浜田聡　② (北海道)桜田雅徳　(福島県)氏家洋一　(宮城県)阿部泰広　(兵庫県)馬渡一幸　(福島県)西山典考　(石川県)井上直也　(北海道)金子洋一　(滋賀県)朽見之男　(埼玉県)吉本志乃武　(東京都)山外健志　③Ⓐ (埼玉県)小笠原茂　(京都府)平尾健二　(奈良県)大倉享　(神奈川県)毛利順　(埼玉県)横山昌弘　Ⓑ (兵庫県)飯塚清澄　(神奈川県)石渡忠弘　(愛知県)神谷尚　(埼玉県)清水美幸　(福岡県)砥上龍也　④ (大阪府)角田輝明　(東京都)斎藤康二　(長野県)中島寿行　(香川県)藤田撰　(東京都)牛山正人　(埼玉県)中田極　(大阪府)野呂康史　(広島県)渡辺忍　(静岡県)石塚昌一　(兵庫県)足立正之　(神奈川県)長嶋雅三　(山口県)藤川和彦　(兵庫県)森田敬太　(静岡県)高木功　(大阪府)岡田信行　(大阪府)片田稔　(千葉県)岡部健悟　(宮城県)岸波徹　(東京都)内山保栄　(東京都)黒瀬直樹　(愛知県)松橋光秀　(群馬県)庭山雅彦　(大阪府)南出昌登　(広島県)小田原育也　(愛知県)伊藤基文　⑤ (山梨県)川口年樹　(北海道)大山進　(山口県)坂田信朗

以上の方々が当選されました。おめでとうございます。

THE SOFTOUCH

TEST 4では，最近発表されたSOFTで，読者の要望の高いものの中から，評価グループがよりすぐって対象としました。

●●●今月のMENU●●●

PART 1
トワード 8
バックマン
スペース・クルーザー
クロスワード英単語
PART 2
フラッピー
PART 3
新作SOFTWARE 27

PART1

TEST4

1. 操作性：操作手順，キー操作など。
2. 画面処理：グラフィック効果，見やすさ，メッセージ処理，芸術性など。
3. 評価段階

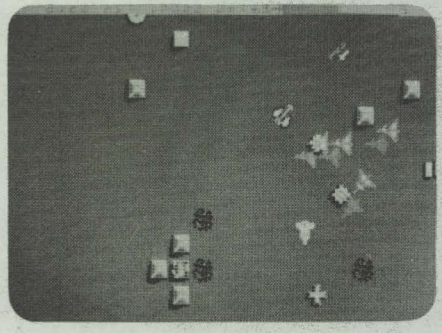
♥♥♥♥♥♥♥	素晴らしい
♥♥♥♥♥♥	よくできている
♥♥♥♥♥	まずまずである
♥♥♥♥	普通
♥♥♥	少しものたりない
♥♥	劣っている
♥	劣悪である

評価グループ
有田隆也　青木秀浩　祝　一平　牛嶋昌和　小笠原吉義　杉浦勇一
清水和人　斎藤　晋　高野庸一　谷川展子　三宅流衣子　Itti Rittaporn

八方美人も楽じゃない

トワード 8
～ 8方向スクロールゲーム
使用機種：X1
必要な機器：G-RAM
メディア：テープ
価格：3,600円
発売元：エニックス
☎03-366-4251

Y.S.	評価項目	R.M.
♥♥♥♥	操作性	♥♥♥♥♥
♥♥♥♥♥	画面処理	♥♥♥♥
♥♥♥♥♥	音響効果	♥♥♥
♥♥♥	独自性	♥♥♥
♥♥♥	物語性	♥♥♥
♥♥♥♥♥	速度	♥♥♥♥
♥♥♥♥♥	熱中度	♥♥♥♥♥



ゲームセンターに「ゼビウス」が出はじめたころ，友人に「おもしろいからやってみな」といわれてやってみると，これがほんとうにおもしろいのです。こんな感じのゲームをパソコンで実現できたらきっとうけるだろうな，と考えていたのです。それが今「アルフォス」，「サンダーフォース」としてたいへんな人気をよんでいます。

知名度はいまひとつのようですが，この「トワード8」もゼビウスによく似たゲームのひとつなのです。

スピードの点では「サンダーフォース」より劣りますが（サンダーフォースが速すぎる？），8方向スクロールになっていて結構よくできています。攻撃を受けてもすぐにこちらがやられるわけではなく，ダメージが100%になるまでゲームを続けられます。ですから初心者でもある程度の点がとれるのです。「サンダーフォース」では難しいという方に向いていると思います。ジョイスティックも使えます。スティックのほうがやりやすいようです。（Y.S.）

サンダーフォースが出たあとだけに，期待は難かしいかなと思っていたんですが，なんのなんの。それなりに趣向を凝らしております。

まず，ジョイスティックが使用でき，熱中のあまりキーボードのたたきすぎで手が腱鞘炎になるというのは防げるようになりました。良かった。ただ，スクロールのしすぎで目が寄れる恐れがあるのは，相変わらずのようです。気をつけましょうね。

画面数は定かではありませんが，シーンによっては，一瞬目の前がまっ暗になったり，画面が突然4分割して目が散ったりと，大変目まぐるしいのですが，オッ！　と思うのはシーン3までで，シーン4からは組み合わせのバリーエーションという感じ。加えて，登場するキャラクタの多さにはびっくりしますよ。まさに自分を見失うほどですからこれはすごい。それにしても，どうしてリアルタイムゲームのマニュアルに書かれている物語というものは，取って付けたようなものが多いのでしょうね。もう少し，この辺を考えてみたいものです。（R.M.）

蘇える現代のヒーロー

パックマン

～ リアルタイムアクションゲーム

使用機種：MZ-700/1200/80K/C

必要な機器：PCG使用可

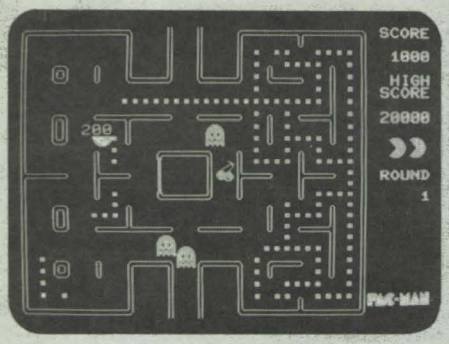
メディア：テープ

価格：3,000円

発売元：電波新聞社

☎03-445-6111

N.T.	評価項目	Y.S.
♥♥♥♥♥	操作性	♥♥♥♥♥♥
♥♥♥♥♥	画面処理	♥♥♥♥♥
♥♥♥♥♥	音響効果	♥♥♥♥♥
♥♥♥	独自性	♥♥
♥♥♥	物語性	♥♥
♥♥♥♥♥	速度	♥♥♥♥♥♥
♥♥♥♥♥♥	熱中度	♥♥♥♥♥♥



パックマンと言えば，もうだいたいのゲーム小僧は知っているはずだろう。アメリカでも大人気を博したゲームで，ゲームセンターには必ずおいてあったし，キャラクター商品も出回っていた。

今回，電波新聞社から発売されたパックマンは，MZ-700，MZ-1200，MZ-80K/Cとどうしてもグラフィック機能で他の機種に劣っていた分を，PCGという周辺装置で，高解像のグラフィックを得ることができ，ゲームセンターにおいてある，あの画面とかわらないようになっている。もちろん，PCGを持っていなくても，ノーマルタイプのプログラムが入っているから，ゲームができればそれでいいという，横着者の少年もいると思うが，せっかくPCGプログラムが入っているので，ゲームセンターの雰囲気にこだわって楽しんでもらいたい。

ゲームの内容についてはあえて説明はしない。ゲームセンターの物とほぼ同じだから。ただ，レバーがテンキーにかわったので，最初はとまどうかもしれないが…。 (N.T.)

ゲームの内容は説明するまでもないでしょう。「インベーダーゲーム」について大ヒットした，あの「パックマン」です。このゲームは，1度始めると思わず熱中してしまう魅力を持っています。最近はゲームセンターでもあまり見かけなくなってしまいました。もう1度思うぞんぶん遊んでみたい人，まだ1度もパックマンをやったことのない人，買っておいて損のないソフトだと思います。

MZ-80K/C/1200でも使え，MZ-700では当然ながらカラーで表示されます。セミグラフィックを使ってパックマンらしい形を出しています。また，PCGを使えばゲームセンターのものとくらべても見劣りしない画面になります。PCG用のプログラムはテープのB面に入っています。

オール・マシン語でできているためスピードも十分です。欠点はというと，やはりある程度続けると飽きてくる点でしょう。「今さらパックマンなんて」といわずに再度挑戦してみませんか。 (Y.S.)

アクションゲームの原点

スペース・クルーザー

～ リアルタイムアクションゲーム

使用機種：MZ-2200/2000

必要な機器：MZ-2000はG-RAM要

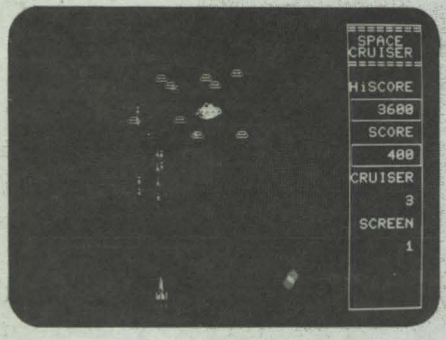
メディア：テープ／5″ディスク

価格：T：4,500円／5D：7,500円

発売元：ニデコムキャリーソフト

☎0963-63-0211

S.S.	評価項目	H.A.
♥♥♥♥	操作性	♥♥♥♥
♥♥♥♥	画面処理	♥♥♥♥♥
♥♥♥	音響効果	♥♥♥
♥♥♥	独自性	♥♥♥♥
♥♥♥	物語性	♥♥♥♥
♥♥♥♥♥♥	速度	♥♥♥♥
♥♥♥♥♥♥	熱中度	♥♥♥



このリアルタイムゲームは，アーケードゲームからの移植版で，オリジナルに少し手を加えたものです。といっても，最近のアーケードゲームの人気は，スクロールゲームまたは，スポーツゲームに集中しているので，これらのゲームファンにとっては，少し物足りないのではないでしょうか。と言いながらも，評価そっちのけで遊んでしまいました。

さて，最初に母船を中心としたＵＦＯ連隊に遭遇しますが，母船を最後まで残しておくと，母船から発射される波動エネルギー砲が，普通の倍の速度となり，これを避けながら母船を撃破するには，一苦労させられます。また第2面では自分が下手なのか，アステロイドゾーンの流星の雨あられの前には，何度か撃墜されました。しかし，よく考えてみると，流星群を攻撃してやられたのだから，この場面は徹底的に逃げるonlyでいい，などと安易な考えから実行したら，ホント簡単にクリアしてしまいました。ただ，得点はアップしませんけど。

全体を通して，敵がもう少し巧みに攻めきてほしいですね。 (S.S.)

ニデコムキャリーソフトブランドの製品で1作目「フロントライン」に続くアーケードゲームメーカー，タイトー社同名ゲームのパソコン映像複製ソフト。前作同様，キャリーラボが移植など技術面を担当し，ニデコが売り出しているもの。テープ，ディスクと媒体の違いにより2種類用意されている。今回評価に使用したものはテープ版であるが，内容はディスク版ともに同一である。オールマシン語でIPLからロードする方式で，キャリーラボオリジナルモニタTS-2000上で動作する。多分BASEやWICSなどで開発したものであろう。思ったより高速で申し分ない。オリジナルのほうを遊ばれた方にはどのようなものかおわかりかと思うが，インベーダゲームの発展上にあるものでいくつかの違ったステージが順にあらわれる。動きは複雑でよく敵を見ていないとゲームオーバーになってしまう。よく見ているとオリジナルに比べアルゴリズムはかなり割愛されているが，プレイ自体に支障をきたすほどではないようだ。初めて遊ぶ人には結構楽しめるし，奥が深いゲームといえそう。(H.A.)

君も英単語の名人になれ

クロスワード英単語

～ パソコン学習ソフト

・中学必修英単語　PART-Ⅰ～Ⅲ

・高校必修英単語　PART-Ⅰ～Ⅴ

使用機種：MZ-700

必要な機器：なし

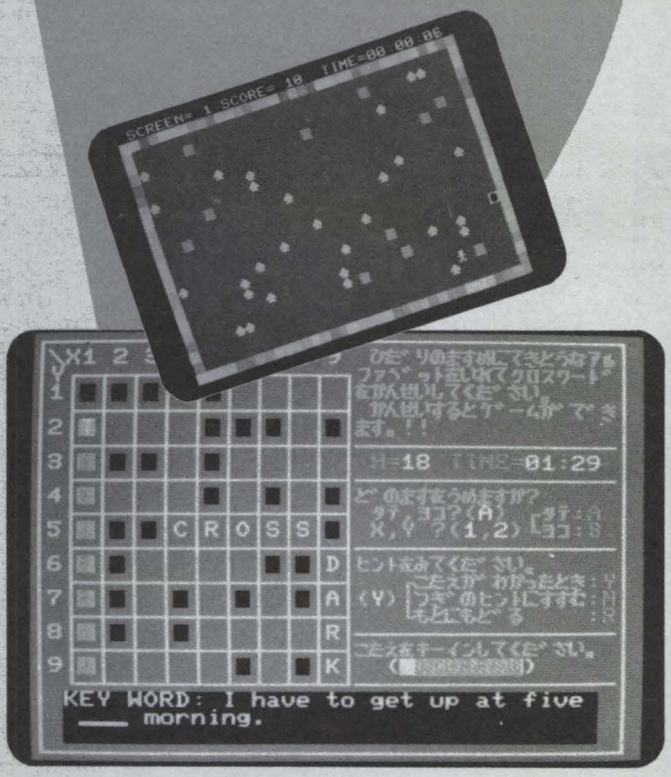
メディア：テープ

価格：各4,800円

発売元：岡山マイコンシステム

☎0862-93-6877

Y.T.	評価項目	R.M.
♥♥♥♥	操作性	♥♥♥
♥♥♥♥♥	画面処理	♥♥♥♥♥
♥♥♥♥	説明書	♥♥♥♥
♥♥♥♥♥	エラー処理	♥♥♥♥
♥♥♥♥♥	実用性	♥♥♥♥♥



英単語クロスワードパズルは，英単語の学習用として，本などにもなっていますが，このソフトはそれをプログラムにしてしまった。しかもその上，クロスワードを完成すると，シンプルなものであるがゲームをやらせてくれるという，なかなかに親切なソフトです。種類は，中学生用がPART-I～Ⅲ，高校生用がPART-Ⅰ～Ⅴまでの合計8種です。それぞれに出てくる単語も大体において妥当ですね。

さて，それぞれ1本のソフトには，9つのクロスワードパズルが入っていて，初段，二段，…八段，名人となっています。段が上になるほど，ヒントを見せる回数が減っていきます。

クロスワードパズルの完成のしかたは，まず，タテかヨコかを，AかBで指定し，次にx，y座標で位置を指定します。このとき第一ヒントが画面の下に表れます。この第一ヒントだけでわかった場合はYを入力して，単語を打ち込みます。わからない場合はNを入力して第二ヒントを見ます。それでもわからなければ，再度Nを入れて，第三ヒントですが，ヒントはここまでしかありません。結局見当がつかない場合は，Rを入力してもとに戻り，別の方向から攻めることになりますね。だたし，あんまりこんなことを繰り返していると，ヒントを見る回数には制限がありますから，注意が必要です。一応全部埋め終わると〝BAD〟か，〝GOOD〟が出ますから，BADの場合は，もう一度よく考えて修正ということになります。どこが誤っているのか教えてくれないのが憎たらしいですね。

単語の位置を指定するのに，座標で打ち込まなければならないあたりは，ちょっと操作性に欠けますが，こんなソフトもあってよいでしょうね。

(Y.T.)

遊びながら学ぼうという試みは，教育関係者たちにとっての理想でありまして，昔から歌にあわせて覚えたりとか，ぬり絵をしながら学んだりといろいろ考えられてきました。その中でもっともポピュラーなのがクイズ形式の問題で，クロスワードパズルなどはまさに本家本元といえましょう。それを，パソコンにやらせてしまったのが，このクロスワード英単語です。クロスワード中学必修英単語PARTⅠ～Ⅲの全3巻。クロスワード高校基礎英単語PARTⅠ～Ⅴの全5巻です。

以前から，教育ソフトとしていくつか各社から発売されておりましたが，必修，基礎と銘打ちながら期待はずれなのが実に多かった。どだい数多い単語数の中から1本ないし2本のテープに収めてしまおうというのが無理なことでして，選択する単語と作るパズルの間にギャップが生じるのはしかたのないことなのでしょう。確かに，マメタン，シケタンを参考にしたとしても，すべてを網羅できるわけではなく，またそれらすべての単語が必修であるわけでもないですから，その周りが教育ソフトを作るうえでの難しいところといえます。

というところでこのソフトは，中学生用3巻，高校生用5巻と多少の物足りなさはあるとしてもよく集めてあり，おまけに，できあがるとゲームができるというのがなけてくるところです。ゲーム自体はとても簡単であるとしても，各巻それぞれ違ったプログラムが入っております。ちなみに，中学生用ではSKY HELPER，バクライ，UFO CRASH，高校生用では，メイロパニック，カーレース，ドッキング，ブロックくずし，マツタケがりなどです。たわいのないゲームはちょうど勉強のあい間のCoffee Breakですね。ただ，キー操作が多少めんどくさいのが気になりました。

(R.M.)

Challenge フラッピー

FLAPPY フラッピー
使用機種：X1/X1C/X1D
価格：テープ版／4,500円
5D，CF版／6,800円
発売元：デービーソフト
☎011-251-7462

圧倒的面数，何と200面をほこるゲームです。世界的にも（?）めったに聞きませんなあ。しかもカセットテープで供給されるのですから。

ブルーストーンをブルーエリアに持っていくことにより1面が終了します。プレイヤーが操作するのはヒヨコのようなかわいらしいキャラクタのフラッピーで，ストーンを押して移動させることができます。押すことだけができるというのはあの11面以降がイヤラシイ『倉庫番』に似ていますね。フラッピーは平面図ではなく横から見た図が画面に示されており，きちんと重力に従って落下します。

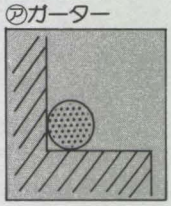
押しては落とし押しては落としを繰り返すことによりゲームは進行します。ブルーストーンのほかにレッドストーンが何個もあり，これをブルーストーンの移動を助けるために使います。このレッドストーンはもろく，壁やフラッピーなどではさまれて押されるとつぶれて消滅してしまいます。このレッドストーンをどのように使うかがミソであり魅力でもあります。

これだけではふつうのパズル型のゲームですが，エビーラとユニコーンがジャマをしてきます。制限時間があり敵がおそってくるために，反射神経型のおもしろさも加味されてオリジナリティあふれるゲームとなっているのです。ユニコーンはそれぞれ一定の高さのところを動いていますがエビーラはどんどんフラッピーを追ってくるので気がぬけません。やっつけるにはストーンを落としてつぶすしかありません。ただし武器として催眠きのこがあり，スペースキーを押すと今向いている方向に飛び出します。これは一定時間だけ相手をマヒさせることができます。

200面と聞くとうんざりしてしまう人も多いかもしれませんが，どの面もそれぞれ個性があり，数だけ増やしたのではないことがすぐわかります。一筋縄では解けないような面も多く，数人がかりで考えなくては解けなかった面もあります。絶対あきらめないでいやというほど考えること，それだけです。高く評価したいものが『パスワー

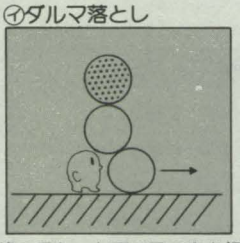
その1 これをやってはダメ 自滅テクニック

㋐ガーター



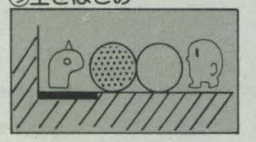
▲石は押すことはできても引くことはできないので青石をすみに持っていってしまうともうダメ

㋑ダルマ落とし



▲積み重なった石の下の方を押し切ってしまうと上の石が落ちてきて，つぶされてしまう

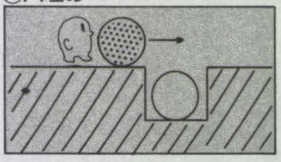
㋒生きばさみ



▲「もうすぐゴールだ！」と気を抜いてゴールと青石の間にユニコーンを入れてしまうと万事休す。押しつけても，やっつけることはできないのだ。

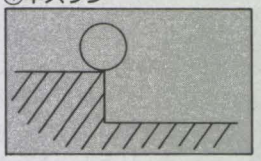
その2 これを知らなくてどうする 基本テクニック

㋐穴埋め



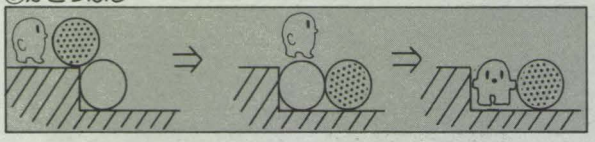
▲青石の通り道に穴があるときには，赤石で埋めるべし。上を青石が通ってもこわれないが押せばこわれる。

㋑半ズラシ



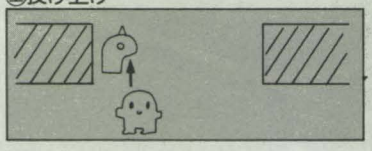
◀石は，石1個分ではなく半個分だけズラすことができる。さまざまな応用のもとになる。

㋒かどつぶし



◀ガーターをさけるテクニックである。まず赤石でかどを埋め，その上を青石を通す。それから赤石を上からつぶせばよい。

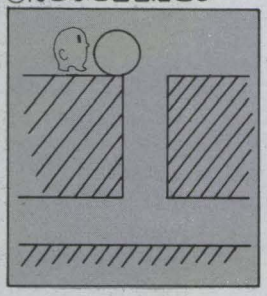
㋓投げ上げ



◀きのこをユニコーンに噴射するときにはなるべく下から行なったほうがよい。なぜならばはずれても，重力の影響で再びすぐ戻ってくるからである。

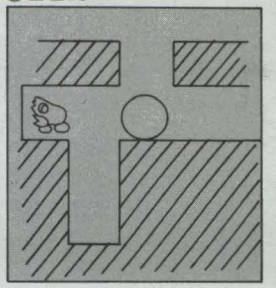
その3 エビーラをこうやってやっつけよう必殺テクニック

㋐待ちぶせ岩石落とし



◀エビーラをやっつける基本的戦術である。すぐ岩石を落とせる状態で待ちぶせしておきタイミングよく落とす。

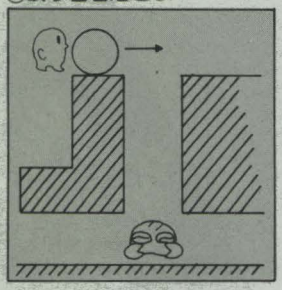
㋑岩石封じ



◀別にやっつけなくても，どこかに閉じ込めてしまえばよいのだ。

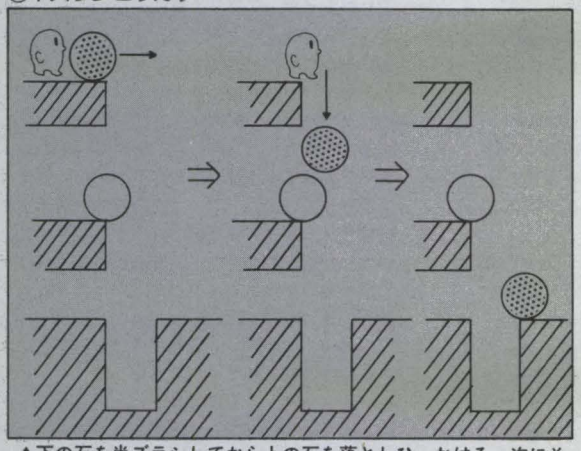
▶うまく，岩石の落下するところにきのこをふきつけて眠らせておいてから，岩石を確実に落としてやっつける。

㋒眠り岩石落とし



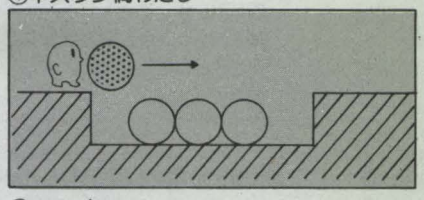
その4 秘密のお話です ㊙応用テクニック

㋐半ズラシヒッカケ



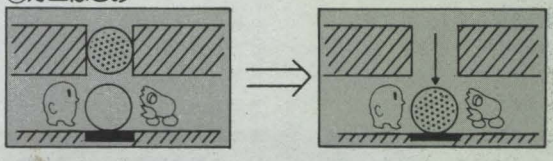
▲下の石を半ズラシしてから上の石を落としひっかける。次にその石を押すと，半個分ズレたところに落とすことができる。これにより，溝をこえたりできる。

㋑半ズラシ橋わたし



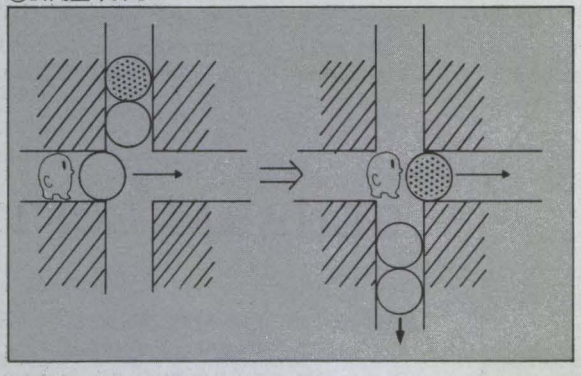
◀石4個分の穴を半ズラシヒッカケなどを用いることにより，3個で埋めてから上を青石を通過させる。

㋒カニばさみ

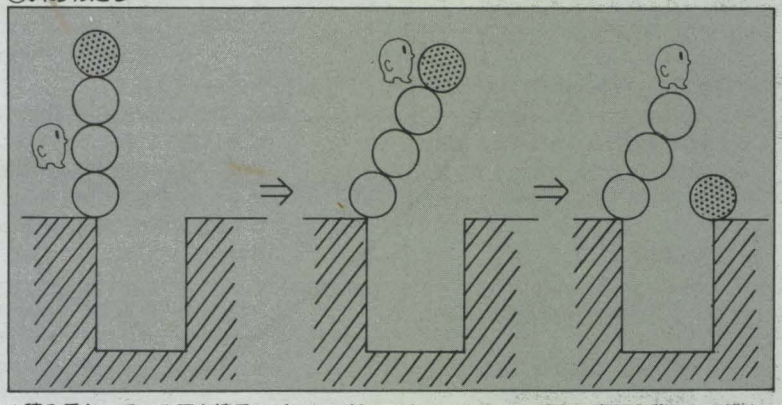


◀エビーラをひきつけておき，その間に赤石をはさみつぶし，青石を落下させる。これを使わないとダルマ落としでやられてしまう。

㋓瞬間空中押し



㋔斜めわたし



▲積み重なっている石を慎重にズラして斜めにする。そして一番上の青石を落として溝をこえさせる。

◀つっかえている赤石を押した直後，青石の落下に合わせて押し出し青石だけを横に突くものである。

ド』です。これは5面クリアごとにひとつずつ与えられます。これを使えば次に挑戦するときに途中から開始することができるのです。じつにうれしい機能ですね。今までの「大作」と呼ばれるゲームでは，死んでしまうとふりだしというものが多く，もう1度最初からやるのはちょっとというパターンにおちいりがちでした。

私が体得した種々のテクニックをまとめてみましたので，お役立てください。半ズラシの応用法をマスターし，エビーラから自由に逃げまわることができたら案外簡単に面を突破することができるでしょう。このテクニックをすべて身につけ，「このテクニックがのっていないなあ」などといいだしたならば，あなたはもう200面に到達しているかもしれませんね。（有田隆也）

PART3

新作SOFTWARE27

-T- テープ
-5D- 5インチ両面倍密
-CF-コンパクトフロッピー
RT リアルタイムゲーム
S シミュレーション
A アドベンチャー
RP ロールプレイング
Sy システム／ツール
H ホビー
P 実用ソフト
☆ グラフィックス
𝄞 ミュージック
E 教育ソフト
B ビジネス
W ワープロ

GAME SOFT

■X1シリーズ

RP挑戦！ 嵐の小惑星帯 ／ パルコム X1 (G-RAM) －T－ 3,000円 ／ わが太陽系連邦軍特殊部隊は，帝国司令部を目前にして帝国軍の反撃により全滅寸前だった。どうきりぬけるかは，あなたの手にかかっている！

RTトワード8 ／ エニックス X1 －T－ 3,600円 ／ 超科学戦争の真最中。今，栄光への脱出が始まった。X1の持つ機能をフルに活用したリアルな効果音とジョイスティック対応が魅力。画面が8方向にスクロールするSFダイナミックゲーム。

S麻雀狂 ／ VING SOFT X1 (G-RAM) －T－ 3,800円，－5D－ 6,800円 ／ 麻雀ができない方にはコンピュータが指導致します。4段階のレベル設定が可能で，漢字表示（漢ROM不要），リピート機能，リプレイ機能があります。

S信長の野望 ／ 光栄マイコン X1（G-RAM） －T－ 4,500円 ／ 戦国時代，弱小尾張を率い天下統一を夢み，実現した織田信長は戦の天才だ。今，貴方は信長となって各地の猛将と戦わねばならない。果たして貴方は天下を統一できるか。

Sホイホイ ／ 光栄マイコン X1（G-RAM） －T－ 3,500円 ／ 不精な貴方の部屋は今やゴキブリ天国。貴方は押し寄せるごきぶりからケーキを守れるか？ コミカルなパロディシミュレーション。

Sナイトライフ ／ 光栄マイコン X1/X1C －T－ 4,500円 ／ コンピュータが，安全日危険日の判断から，2人の予定プレイタイムに基づき，最適な体位をいくつか選び出しグラフィック表示で同時進行でお教えします。アダルトソフトの代表作。

RTフラッピー ／ デービーソフト X1 －5D，CF－ 6,800円 ／ 大好評のフラッピーのディスク版。

■MZ-700/1200/K/C

AAUTOCRACY／ BASIC HOUSE MZ-700 －T－ 3,800円 ／ ある朝，あなたは起きてすぐ昨日見た夢を思い出しました。それは不思議な夢で，地球のどこかで独裁政治が行なわれていて，そこに住む住民たちがたいへん困っている…という夢でした。そして…

Sタイムシークレット ／パスカルII MZ-1200/K/C －T－ 3,300円 ／ すでに圧倒的な評価を得ているタイムシークレットの1200/K/C版登場。

■MZ-2200/2000

Sナイトライフ／ 光栄マイコン MZ-2200 －T－ 4,500円 ／ コンピュータが，安全日危険日の判断から，2人の予定プレイタイムに基き，最適な体位をいくつか選び出しグラフィック表示で同時進行でお教えします。アダルトソフトの代表作（カラーBASIC用）。

■MZ-5500シリーズ

S北アフリカの嵐／ Brainhill MZ-5500－5D－ 7,800円 ／ 北アフリカ砂漠を縦横に疾駆する名将ロンメルに対して，英軍は大攻勢に転じた。一方ロンメルにとって，頼りになるのは一握りのドイツ・アフリカ軍団の精鋭のみ！ 物量を誇る英軍をたたきつぶし，ふたたびアフリカ砂漠を手中におさめることができるだろうか。今，北アフリカに鋼の嵐が吹く！

S外宇宙艦隊を撃破せよ ／ Brainhill MZ-5500 －5D－ 7,800円／太陽系を開発し，繁栄を謳歌する地球人。しかし，秘かに侵略者は近寄っていた……今，ナゾの敵は外殻基地を破壊し，太陽系の中心部へ迫ろうとしている。強大な外宇宙勢力を前に，綿密な作戦と指揮が人類の運命を分ける。果して地球は救われるか。

Sモール特攻作戦／ Brainhill MZ-5500 －5D－ 7,800円 ／太陽に向け，凄まじいスピードで進撃してくる恒星破壊兵器〝モール〟。迫り来る危機から太陽を救うには，コマンドチームがモールに進入し，制御システムを破壊するしかない。太陽消滅を目前に，特攻作戦は始まった。コンピュータ・ストラテジーゲーム。

Sウルフパック／ Brainhill MZ-5500－5D－ 7,800円 ／ 大西洋に出没するUボートの驚威は，宿敵イギリスを恐怖のどん底に陥れた。目標は航行するすべての敵艦船，そしてイギリス海上補給路の寸断。戦果はいかに！ 貴艦の幸運を祈る。

インペリアル フリート

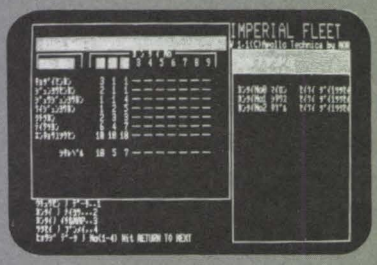
戦闘シーンも鮮やかに映像化したウォーシミュレーションゲーム。

ときは西暦2821年，プロキシマ星系の領有をめぐって，銀河帝国と地球連邦の対立はいよいよ激化してきた。銀河帝国の総督であるキミは，皇帝の命を受け，地球連邦との闘いに速やかに勝利しなければならない。

はじめに，11もある星系の各惑星の中から敵を捜し出す。敵と遭遇すると，戦闘に突入するか否かの判断を迫られる。自軍艦隊の詳細なデータをチェックし，彼らの力量を冷静に判断しなければならない。戦いに入ると，臨場感あふれる戦闘音が響き，戦闘シーンが展開される。戦績が思わしくないと，皇帝から戦費を削減されたり，戦艦の建造を禁止されたり，あげくの果ては指揮官不適格のレッテルを貼られてしまうので，うかうかしてはいられない。

壮大な宇宙を舞台にしたウォーゲームとして，知的スリルを満喫させてくれる。

X1用 テープ版 3,300円 発売元：
Apollo Technica ☎06-32-0555



GENERAL SOFT

■MZ-700

Eクロスワード社会－中学歴史年号編－／ 岡山マイクロシステム MZ-700 －T－ 4,800円／中学歴史の重要な年号を使用してスロスワードを完成する。完成すると「ミサイルコマンド」というオリジナルゲームができる。クロスワードには初段から名人までのパターンがある。

■X1シリーズ

独占!! 情報管理／ 近畿コンピュータサービス X1 －5D，CF－ 15,000円 ／ プレイ情報，ビジネス情報を管理し，必要に応じて，検索する条件によって該当する情報を画

面またはプリンタで出力できる。メール用シール，ハガキの宛名印刷機能がついているので各種催物，見込客，会員などの案内状送付の宛名書きが簡単にできる。

実戦 // 在庫管理／ 近畿コンピュータサービス XI －5D，CF－ 15,000円 ／ スモールOA用。在庫金額1億円未満。在庫数量100万未満。仕入・販売管理の商品元帳データの共用ができ，入庫・出庫伝票発行機能付の他，商品在庫一覧表，適正在庫過剰一覧表，適正在庫不足一覧表，入出庫日計表，入出庫月計表などがプリンタ出力できる。

実戦 // 販売管理／ 近畿コンピュータサービス XI －5D，CF－ 15,000円 ／ スモールOA用。月商・在庫金額1億円未満。商品在庫数は100万未満。仕入・在庫管理の商品元帳のデータの共用ができ，入庫・売上げ伝票の発行機能の他，請求書発行，発行先名簿，商品別売上げ日計表，月計表，得意先別売上げ月計表，入金日計表，その他ができる。

漢字顧客台帳 ― 漢字版宛名書パッケージ／ 大江 XID（要漢字ROM）－CF－ 19,800円 ／ 顧客No.，氏名，キーワードの項目の高速サーチ，その他全項目のサーチ可能，漢字検索可能。カナ漢字変換，熟語登録語500語可能。顧客データ最大登録件数800件。今日から使えるタックシール1000枚入

■MZ-5500シリーズ

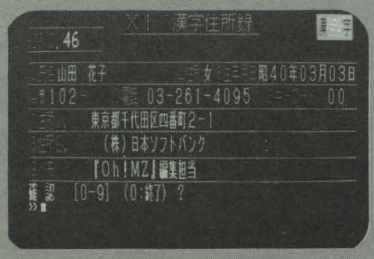
SUPER PRO 販売，仕入，在庫管理 ／ リードレックス MZ-5500 －5D－ 200,000円 ／ ■売掛金管理，買掛金管理，粗利管理，在庫管理。■締日には請求書を自動発行。■棚卸しも簡単。■各種日報，管理資料も豊富。■漢字で表示・印刷。■売上伝票，請求書は(株)日本法令製を使用。■画面との対話型入力でやさしい操作。

SUPER PRO 財務会計 ／ リードレックス MZ-5500 －5D－ 250,000円 ／ ■仕訳入力だけで決算まで。■月 4,000件までの仕訳が可能。■勘定科目は358。■内訳設定で補助簿作成。■摘要は2タイプ。■漢字で表示印刷。■二重の機密保護機構

SUPER PRO 給与計算 ／ リードレックス MZ-5500 －5D－ 200,000円 ／ ■月例給料，賞与から年末調整まで。■時給者，日給者も処理可能。■銀行振込の依頼も簡単。■(株)日本法令の帳票を使用。■漢字で表示・印刷。■変動データの入力だけで給与明細書。■支給対象者は300名まで。■支払額の端数整理ができます。■機密保護機構

実戦仕入管理 ／ 近畿コンピュータサービス XI －5D，CF－ 15,000円 ／ スモールOA用。月仕入金額・在庫金額1億円未満，在庫・販売管理の商品元帳のデータの共用ができ，仕入伝票，出庫伝票発行機能の他，商品別仕入月計表，支払い日計表，仕入先別仕入日計表，仕入先別仕入月計表，買掛け残高表，商品在庫一覧表などをプリンタ出力できる。

漢字住所録



X1シリーズとしては初の漢字住所録が発売されたということで，サンプルを頂きましたので，簡単に紹介したいと思います。

システム構成は，X1D（今のところCFだけです），拡張I/Oポート，漢字ROM，漢字プリンタが必要です。

データ項目は，No.（1～800まで自動的に作成），氏名（10字以内），性別，生年月日，郵便番号，電話番号，キーワード（00～12までの分類登録），住所1，住所2（各20字以内），メモ（20字以内）の9つで，全項目についてサーチ可能，うちNo.，氏名，キーワードは高速サーチ可能となっています。

漢字入力は，カナ漢字変換と熟語登録機能（500熟語まで登録可能）が使えます。

印刷は，全データ印刷と宛名書き印刷それぞれに検索印刷が可能で，パーソナルユースの他一般小売業のユーザー管理や，DM発行などに使えます。

XID用3"CF版 19,800円 大啓
☎052-851-7253

HP王将

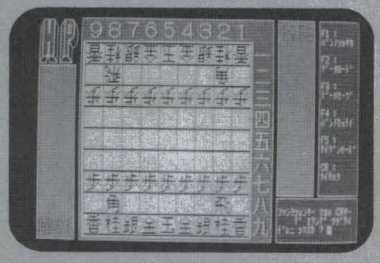
注目の「王将」がグーンとバージョンアップ，名づけて「HP王将」として登場。

将棋指しにとって，木の盤にぴしゃりと駒を置く快さは軽視できないものですが，そのあたりの感じを出そうとした工夫がしのばれてくるのです。見やすくレイアウトされた画像の中央にきちんと置かれた，黄土色の盤面を前にすると，こちらも思わず姿勢を正したくなります。駒の移動も盤上のカーソルを動かすので，〝指す〟感覚を十分味えます。

パソコンとの本将棋で難点のひとつとされていたのが，コンピュータくんの長考。しかし，思考時間も，かなりスピードアップし，イライラせずに済むようです。

なによりうれしいのは，盤上への駒の設定が自由にできること。名人戦を再現して途中から自分で指したり，駒落ちや詰め将棋も自在なので，楽しみの幅が広がります。

X1用（要G-RAM），MZ-2200/2000用（要G-RAM1.2.3），テープ版，各4,200円
マイコンハウスSPS ☎0245-45-5777



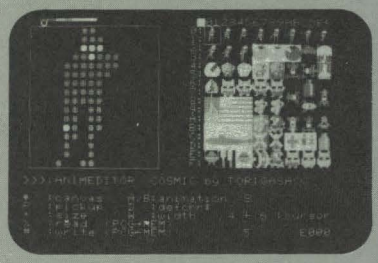
アニメディター

X1の持つPCG機能を生かすためのエディタ。作成したデータは，あらゆる応用を考慮し，マシン語データとして保存，BASICで取り出すことができるという汎用性を有する。プログラムはマシン語でHuBASICのモニタからロード，その基で動作する。サンプルデータが付属。

HuBASICではDEF CHR$でこのPCGをサポートしているが，あくまでも領域にデータをセットするだけで，エディタ機能はない。つまり，このDEF CHR命令を使うためには何らかの手段により，フォントをデザインしデータに変換してやらなければならない。こうひと言で表現できるが，これを手で行なうと実にしんどい。

価格が5,800円と少々高いような気がするが，あって不便なものでなく，ゲームキャラクタをデザインしたりする場合非常に有効なツールといえよう。処理もマシン語であるため高速。申し分ないと思う。操作性も十分で必要なキーインストラクションは常時ディスプレイされているが，最初使用する際は説明書にひととおり目を通しておくことをお推めする。モードは4つありキャンバス，エディタ，パレット，アニメーションの各モードからなる。デザインは8×8あるいは4キャラクタ分16×16ドットのいずれでも可能。大きなキャラクタをデザインする場合16×16ドットモードは実にありがたい。そして，特徴の1つであるアニメーションモードは次々とキャラクタを写し，動きをチェックできるというもの。PCGを使ったゲームなどは，この種のチェックが動きを安定させるために十分必要であろう。

X1用テープ版 4,500円 発売元：ダイツー／COSMIC SOFT ☎06-344-5150



◆レイアウト担当の中島真子さん。あなたは独身ですか？　私は石野真子のファンです。
日比野　泰彦(24)愛知県

◆「僕のいちばん苦しい時間……」
カップラーメンにお湯をつぎ，ファンヒーターのスイッチを入れ，MZ-2200 の BASIC を立ち上げる(さむ～い，ひもじい)。なぜ，みんな，２分も３分もかかるんだ！　１分でできるとか，一発点火だとか近ごろあるのにMZ-2200は……。やはり，ディスクしかないのだろうか。
北野　利幸(14)福井県

◆MZ-700でCP/Mを使いたい。何とかしておくれ～。
山後　誠司(27)熊本県

◆はたして，MZ-700用のCP/M は発売されるのかな。RS-232C I･Fカードも出てほしいし，と考えているうちに，また夜が明けました。
小林　雅彦(22)茨城県

◆MZ-700でFDが使用できない，とSTUDIO MZ によく書かれておりますが，私は自分で工夫して，MZ-80K/C用のFDおよびプリンタを使用しております。
向井　康史(35)愛知県

◆I.FORTRANを勉強したい。
2.だからFDとCP/Mが欲しい。
3.だから１日も早く，シャープさんが MZ-700 用FDとCP/Mを発売してくれることを祈る。
4.でも，国家試験には間に合わないだろうな…。
吉田　誠祐(24)京都府

◆なにがフロッピーディスクだ。カセットだと，トイレに行けるし，その間，ジュースだって飲めるんだぞ～。
だけど，フロッピーディスク欲しいよ……。
東田　充紀(14)奈良県

◆はやくフロッピーディスクを出してほしいな。あんまり待たすと，おこるのだもの。
藤野　清文(17)岡山県

お待たせしました。というわけで，今月号のトピックス，MZ-700用のFDの記事読みましたか？　FD-700もRS-232Cも用意されますよ。

◆THE SOFTOUCHがとてもいい。２月号のタイムシークレットを見て，すぐに買ってしまった。解けたあと３月号を見たら，「内緒話」がわかってとてもとてもおもしろかった。
福島　栄司(17)神奈川県

◆初めてやったアドベンチャーゲーム「タイム・シークレット」が４日間（８時間）で解けてしまった。もっと難しいと思っていたのに残念だった。もっと難しいうんとうなるようなアドベンチャーゲームを紹介してちょうだい。
金居田　陽一(17)栃木県

◆毎月買っています。なんとなくMZ-700のゲームが減ってきていると思うのです。
清水和人さんへ
ぼくもタイム・シークレットに熱中しています。ほんとに解けたのですか？　すごいですね!!　ぼくも頑張ります。でもほんとに解けたんですか？　キイロイハナが手に入らない。城の王様にもらうのかな？　ほんとに解けたの？
大槻　厚史(14)埼玉県

キミも疑い深い人ネ。清水和人といったら，泣く子も黙るアドベンチャーの鬼なのだ。

◆２年前に私がパソコンを買おうと考えていたころ，主に幅をきかしていたハードがPC-8801 と MZ-80Bでした。そこに出てきたのがMZ-2000，でも２もなく購入して１年半。PCは全機種MKIIは出る，富士通はFM-7で一気にシェアを伸ばし，シャープもXIとMZ-5500を世に問うています。

だいたい，ハードを買うときは，カタログを見て選ぶ人がほとんどでしょう。しかし最近，私は悟りを開きました。パソコンを買うのにカタログは無用，値段がわかればだいたい性能の見当はつきます。予算が決まったら，パソコンのコーナーに行かずに，ソフトの棚に直行しましょう。そこから先はもう言いたくありません。昔の栄光はいずこで，MZは地盤沈下，幅をきかしているのはPCシリーズ，FM-7，XIの３機種とMSXぐらいで，その他のマイコンは「マイナーコンピュータ」になり果てています。MZを目前にして思うことは，とにかくソフトが出ない，ということです。

よく，隣りの芝生は青いなどと言いますが，ソフトに関しては明らかにPCとFMに偏っています。実際，ソフトとハードは車の両輪で，よいソフトがあるとハードが売れる，ハードが売れるとソフトも売れる，ソフトが売れると新作が増える，新ソフトが増えるとハードも売れる，という具合いだと思うのです。しかし，今の状態は何ですか。某ソフトハウスの如く，80BやXI からの移植でお茶をにごしているところもあれば，とっとと8801やXIに乗り換えたところとか最初から完全に MZ無視を決めこんでいるところとか，ほんとうに私は，MZ ユーザー××万人になり代って怒っとります。

それと，上位バージョン品のサポートにも困った点があります。オールBASICのソフトだとリストをいじる程度ですみますが，マシン語ではそうもいきません。せめて差額交換ぐらいの誠意を見せてくれてもいいと思うのです。

とか言いながらも，私は自分のマシンを手離す気はありません。ときどき考えるのですが，MZ-2000が発売されて１年８カ月，MZ-2200とはハード的には同等ですから，ハードは十分研究されているはずです。DOSもあります。C-DOS(WICS, BASE)，dB コンパイラもあります。CP/Mも一応走ります。それなのになぜソフトが出ないのでしょう。あとはソフトハウスの熱意のみじゃないですか。とにかく全国のMZユーザーは，良いソフトを渇望しているのです。

以上，言いたい放題でした。
須山　隆雄　兵庫県

FROM READERS

編集部から読者の皆さんへ
お便りをお寄せください。内容はどんなものでもかまいません。言いたい放題↗

◆最近Oh！MZに助けられている。というのは，受験で忙しくMZ-721にさわれないので，一時は愛機を売ろうかとさえ考えたのだが，毎月Oh！MZ を買い，記事やプログラムを見て，また意欲がわいてくるからだ。だからOh！MZは私の愛機MZ-721の命の恩人だ。
端　司健(16)大阪府

MZ-721だけの命の恩人じゃなくて，キミにも感謝されたい。大学に受かってネ。

◆MZ-2000を手に入れて約１カ月。何もわからない私は，難しすぎる(!?)マニュアルを相手に悪戦苦闘の毎日。だから，３月号の「BASIC入門」はとてもありがたく思うのです。
田中　令子(17)東京都

「BASIC入門」は数々のアプリケーションをつけて，今後も皆さんのよい手助けとなるよう頑張ります。

◆もうMZ-80K2/K/Cは活きていないのでしょうか？　私の愛機は蓄積されたソフト（各種記事）を大事につまみながら働いています。今夏からサイボーグ化の計画も……。K/Cユーザーよ，頑張ろう。
山口　昌彦(27)熊本県

サイボーグ化が成功したら，ぜひ投稿してくださいね。待ってます。K/Cユーザーは健在なり。

◆わかった！　文のはじめに"◆"を入れると載るんだ。え，うそだろうって？　ちゃんと載ってるじゃないか。キミもやってみよう！
両津　勘吉(14)岩手県

当たり～。でも，わざわざ◆を付けてきた人はキミが初めてだよ。

◆Oh！MZは，編集者の写真を載せませんね。ぜひ女性の方だけ，写真を載せてください。
小杉　寿文(16)奈良県

当編集室には，女性とおぼしき編集者はおりません。一応，姿形は女性……かな？　と思わせるようなのはおりますが，いや，あれが女性だとは思えんのです。あしからず。

◆今月号のMZ-700のライディーンの方法を使うと，X1で4,5重和音を出すことが可能になりました。感激!!
夏目　昌樹(14)富山県

◆3月号の表紙の左下の絵，あれは〝とうもろこし〟ですか？
坂本　哲也(13)愛知県

◆最近，ポケコンの記事が少ないな，と思っていたら，2月号から「ポケコン横丁」という記事が載りましたね。これにはとてもうれしかったです。友だちはX1やMZ-721，JR-200やPB-100 を持っているのですが，かれらにポケコンでもGRAPHICができるということを納得してもらえました。
青木　和夫(12)茨城県

◆最近X1を買いまして，それに合わせてOh！MZを買いました。BASICのこともよくわかりませんので，遊歩氏のBASIC入門を興味をもって読ませていただいております。X1用のゲームもプロボーリングとバイキンをコロセなど，早速打ち込んで楽しんでおります。
城間　盛和(21)沖縄県

TO THE EDITOR

↗ユニークな内容のものをお待ちしています。あわせてイラストも大募集。STUDIO-MZまでどんどんお送りください。

◆最近，6歳のおいっ子に「おばちゃん，パソコンちょうだい」とせまられ，バリケードでも築こうかと思ってます(ジョーク，ジョーク)。去年，あの夏の暑さを耐えてエアコンがまんしてパソコン買ったんだから。今年は，X1が熱をださないように，エアコン貯金を始めました（ダンナ様も協力おねがい)。
松田　紀子(25)兵庫県

私なら，パソコンがまんしてエアコン買うね。なんちゃって。両方とも手が出ない私なのです，実は。

◆私の部活では，18人中9人がパソコンを持っています。やはり卓球部は暗い。え～い，MZ-700は負けんど～。
北山　陽一(14)兵庫県

パソコン持ってて暗いという時代は過ぎたのです。やっぱ，卓球やってるから暗かったりして……。

◆どうしてMZにはセントロインタフェイス，フロッピーインタフェイス，RS-232Cインタフェイス，GB-IBインタフェイスが内蔵されていないのか？ また，SHARP BASICやHu-BASICなど，それらをコントロールする命令がないのですか？ SHARPさん，別売りなんて……。はたしてここまでクリーンにする必要があるのですか？ 最近少しずつ使えるようになって，疑問や不満を感じるのです。メーカー側は，3年先のマシンを考えているというが，私の意見もとり入れて！
潮見　崇(21)神奈川県

ユーザーあってのメーカー。シャープさん頑張ってください。

◆え!? 「シャープ」って13年前はシャープじゃなかったの？ 朝，新聞を読んでいたら，シャープは昭和45年に社名を「早川電機工業」からシャープに改めた，と載っていてびっくりしてしまいました。ずいぶん対照的な社名だったんだな。
水井　彰(13)神奈川県

え！ そうだったの。実は私も知りませんでした。

◆Oh！MZを買いにいったら，大学生協のおじさんに「工学部の方ですか？」と聞かれたり，1年前では珍しかったパソコンおねーさんも，今ではすっかり市民権を得ました。MZ-700は女性のユーザーもわりと多いのかな。

峰岸氏のすすめに従い，タッチタイピングの本を買ってきたものの，ピアノの先生があきれるぐらい指が短いため，CR や SHIFT が大の苦手。体の大きい欧米女性＆日本男性がうらやましい。ちなみに，私は産婦人科の医師は無理だそうです。この指では！（なんのこっちゃ）
平井　真理子(23)宮城県

そうなんですか？ 指の短い人は産婦人科の先生になれないって。ところで，あなたは何の先生になるんですか？ 女医にパソコン，力強い限りですね。

◆パスカルの記事のおかげで，やっとパスカルがまともに使えるようになりました。今までマニュアルがなかったので，アドベンチャーゲームしてました。コマンドのわからない言語は，へたなアドベンチャーゲームよりおもしろいのです。
片山　淳(22)福岡県

うん，なかなか良いことをいいますね。ひとつずつコマンドを解析していく喜び，そうアドベンチャーゲームに似てるかな。

◆X1用のCP/Mを買った。さっそく起動してみると，画面下にファンクションキーの定義が表示され，「ちきしょう，PCみたいだ」と思ったが，SHIFT ＋ ㊉ でやっぱりテレビ画面がスーパーインポーズされ，「よしよし，X1だ」と納得した。
小沢　健次(24)長野県

◆私の高校には，推定65才の数学教師，N先生がいます。すっごく賢いおじいさんですが，NHKのテレビ講座でマイコンに魅せられたらしく，授業中にもマイコンの話が飛び出し，みんなは時間のつぶれるのを喜んでいます（ちなみに，先生のはカシオの製品だそうです)。
西田　慎二(16)大阪府

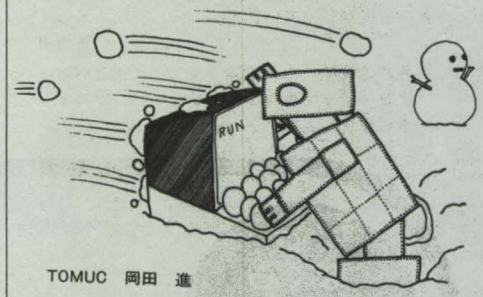
思い出すなあ。余談の好きな先生，みんな大好きだったもんね。

◆先日，超久しぶりにゲームセンターなるものに行った。高得点のだせそうなオーソドックスなテーブルゲームに腰を落ちつかせ，100円投入!! But，テンキー操作に慣れきっていた右手は，スティックの操作に対応しきれず，ついにHi-Score は出せなかった。自信あったのになあ～。
好川　一明(17)奈良県

ジョイスティックを購入してから，再度トライしてみよう。

◆突然ですが，質問があります。それは「競馬予想」のソフトは発売されているのに，「競艇予想」のソフトは発売されていないのですか？
肥田　純治(16)愛知県

キミが競艇やるのかな？ それともお父さんのために探してるのかな？ やはり，競馬よりは競艇のほうがなんとなくマイナーそうだから。だけど，アルゴリズムは同じじゃないだろうか……。



TOMUC　岡田　進

◆2月号から続いているポケコン移植テクニックはすばらしい！ 今度は1250シリーズの機械語の特集をしてください。とにかくこの記事，できる限り継続してください。
根本　洋二(16)茨城県

◆STUDIO MZに出ている人がプレゼントをもらっているのを見たためしがない。だから，載せないで，プレゼントだけください。
指原　淳志(14)兵庫県

そんなことはありませんよ。公平に抽選しているのだから。しかし，キミも暇ですね。

◆すっすごい！ 超高速ペイントルーチンはすごい！ もうFM-7なんか欲しくない！ X1には無限の可能性があるのだ。
児島　敬史(21)福岡県

来月号あたりで，またもや，超・超高速ペイントルーチンを掲載予定です。お楽しみに。

◆私の壮大な計画について少々。毎月Oh！MZを買うだけで，あとは貯金しまくります。すると，来年の受験までになんと〇万円も貯まります。そこにお年玉を加えると，なんと〇万円になります。そのお金に，大学に入ったときの入学祝いを加えて，カラーモニタと拡張I/Oポートとライトペンを買うのでした。終わり。
石塚　成宏(17)兵庫県

もし大学に入れなくて入学祝がもらえなかったら何が減るのかな？ まあ，そんなことのないよう，娯楽投資はOh！MZのみにして精進しようね。

◆ショートプログラムはいい企画ですね。これからもたくさん載せてください。
椎名　英喜(19)千葉県

ショートプログラム大募集中です。誰にでも1本は隠された名作があるんじゃないかな。ぜひ投稿してね。

◆私はあらゆるパソコン雑誌を買って読んでいますが，今だにXIDをパソコンとしてではなく，ただのTVとしか使用できないアホです。もしプレゼントが当ったら，LOADぐらいはできるようになりたいのです。BASICはわからん！ マシン語は簡単そうなのでやる気が起こらん！
岩城　史如(20)北海道

いやあ，その簡単そうなマシン語にキミが熱中するのも間近いかな？

◆XICを買って2カ月。早く使いこなせるようになりた～い。今，XICで松田聖子のヒット曲を聞こうと，日夜頑張っています。

安田　哲幸(25)香川県

それは簡単です。X1をテレビにして，ベストテンにチャンネルを合わせて……。ごめんなさいね。

◆MZ-700を買って約1年。Hu-BASICとS-BASICの使い分け（コマンドの違い）をほぼマスターし，父親の仕事のプログラムを作ってやり，システム開発費をしっかりいただいた僕です（父は国鉄です）。

松尾　善良(21)北海道

赤字続きの国鉄と聞きますが，いくらもらったのかな。Oh！MZの稿料よりはいいのかな。

◆Puzzle BOX3の答6本

編集室のみなさん，このアンケートを読み終わったら，まことにご迷惑でしょうが，Puzzle BOXの係までわたしてください（これで一石二鳥）。

松本　隆(16)鹿児島県

あのな～キミ。それじゃ，40円使ってパズル係にハガキ出した人に悪いんじゃない？

ぼくらの掲示板

売ります

★MZ-5511（RAM256K）＋漢字ROM＋マウス＋漢字CP/Mで27万円（多少応談）〒またはTELで。〒969-04 福島県岩瀬郡鏡石町大字久来石字小栗山476 熊谷光博 ☎024862－6605

★MZ-2000＋G-RAM1,2,3＋ソフト＋カラーBASIC＋書箱＋マニュアル箱を，8万5千円～10万円で。また14型モニタ1D09を6万9千円でハガキを待つ。〒386 長野県上田市常磐城6-4-5 市瀬常在

★MZ-80B＋G-RAM1,2＋MZ-1P07＋ITEM disk 80B＋F-DOS＋ソフト多数。すべて取説，ケース有。20万円（相談に応ず）。〒468 名古屋市天白区御前場町127-8 日比野泰彦

★MZ-731＋マニュアル＋HuおよびS-BASIC＋プリンタ用紙3巻＋インク＋ゲームソフト1本を6万円。S社12"グリーンCRTを2万円で。必ずハガキで。〒237 横須賀市湘南鷹取1-6-8 誉田浩之

★PC-1500＋CE150＋4KRAM＋カナテープ＋ソフトテープ(2本)関連書籍6冊を4万5千円で。往復ハガキで。〒120 東京都足立区千住桜木2-15-1-1102 高橋美和子

★MZ-2200用キーボードカバーを2千円で。まだ新品です。往復ハガキ待つ。〒581 大阪府八尾市東町1-36-7　園田秀章

買います

★PC-1401を1万円で。CE-124付で千円追加。まずは〒で!!　〒466 名古屋市昭和区福江1-5-18 枡田浩義

★シャープのドットプリンタMZ-80PK5を譲ってください。インタフェイスユニット（K2E用）があれば尚良いのですが，連絡ください。プリンタだけでも結構です。〒344 埼玉県春日部市藤塚2117-442　米田雅雄

★MZ-2000用320K RAMボード PIO-3034（I・Oデータ機器）を4万円で譲ってください。まずは往復ハガキで。〒191 東京都日野市西平山2-8-53 石原克己

★PCG-700＋NEW PCG-AID ＋取扱い説明書＋etc. まともに動くもの1万5千円で。多少相談の余地あり。往復ハガキ待ってます。〒963 福島県郡山市愛宕町6-8-302 橋本賢

交換

★当方，MZ-731＋ソフト。貴方，PC-6001MKII。往復ハガキで連絡してください。〒716-13　岡山県高梁市巨瀬町159-3 石川健治

★当方，MZ-2200＋カラーCRT＋レコーダ＋ソフト2本。貴方，X1または13万円で。ハガキ待ってます。〒368 埼玉県秩父市大畑町5-4　茂木義己

仲間

★"VIVA X1 クラブ"設立。日本全国にX1のワを作りましょう。X1は最高の機種ですが，近くでユーザーを探せません。友だちをたくさん作りたいと思います。〒365 埼玉県鴻巣市本町3-12-7　堀内慎一

★"X1ユーザーズクラブ"はソフトハハードの情報を中心に，月1度の会誌を発行！　詳しくは60円切手同封の上，〒805 北九州市八幡東区日ノ出1-17-9 荒武優まで。マシン語研究もします。入って損はないよ!!

★ガンコにも固持し続ける全世界のMZ-700オーナーの皆さん。情報交換しましょう。ハガキまたはTELで！　〒240 岩手県北上市大提南3-4-5-43 岩渕新一　☎0197－67－3039

★MZ-700ゲームクラブの会員募集中！　往復ハガキで連絡してください。〒567 大阪府茨木市太田3-11-1　宮本孝之

PC－5000のハッカーを求む！

MS－DOS、良質の日本語ワープロ(この文もPC－5000でいきなり打っている。Oh！MZ　四月号を見よ。)、表言語、通信用ソフト、GW－BASICがあり、他に何が要るのかと怒られそうだが，この凄いハードを何とか120%使いたいのだ。しかし私には時間が無い。PC－5000のハードを調べている人はぜひOh！MZに発表してください。

岡田　好一(25)大阪府

◆掲載ご希望の方は，官製ハガキに項目(売る・求む……)を明記してお申し込みください。

◆ソフトの売買，交換については，いっさい掲載できません。

◆取り引きについては当編集部では責任を負いかねます。

◆応募者多数の場合，掲載できない場合もあります。

16bit Personal Computer Magazine

Oh!16

新雑誌創刊!!

16ビットパソコンのハードおよびソフトの総合活用をわかりやすく紹介する実用情報誌、登場。

創刊0号の主な内容

特別企画

各社の16ビット主力マシン総覧

- ■ソフトユーザーのためのテクノ講座「dBASEII実践教室」
- ■マルチプランによる経営シミュレーション
- ■座談会／16ビット時代の可能性
- ■DOSライブラリー
- ■パソコンによるデータ通信実験室
- ■16ビット用ソフトのオペレーティングレポート
- ■ザ・ゲーム（16ビットベストゲーム集）
- ■新作ソフトライブラリー
- ■ニューマシン情報

創刊0号は6月発行予定

予価480円(隔月刊)

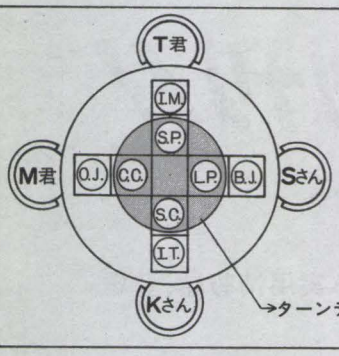
本誌は当出版部発行の季刊「ザ・ソフトバンク」が誌名、体裁、内容を一新して誕生する16ビット総合情報誌です。ソフトに関する前誌の蓄積を引き継ぐとともに、新しくハードおよびOS活用を含む総合的な新情報誌に生まれかわります。前誌のご愛読に心から感謝します。また、今後も新雑誌への倍旧のご愛顧をお願い申し上げます。

日本ソフトバンク出版部　発行／☎03(261)4095

Puzzle BOX5

今世紀最大の名作「スチュワーデス物語」も終わってしまい，編集室も話題にとぼしい毎日ですが，これでようやく落ちついた春の日々が送れるというものです。しかしここでまたみなさんには，ボーッとした頭を混乱させるためにも今月のパズルに挑戦してもらいたいと思います。

暖かくなってきましたねえとばかり同志を集め（本当は季節を問わないのだが）出かけることにしました。私（T君とします）とM君，Kさん，Sさんの4人で，もちろんCAKEを食べに行ったのです。4人のオーダー品は下に示すとおりですが，お店のウエイトレスさんは，PUZZLE BOXの問題になるようにといって，図のように間違った置き方をしたのです



T君	オレンジジュース	O.J.
	チーズケーキ	C.C.
M君	アイスティー	I.T.
	ショートケーキ	S.C.
Kさん	バナナジュース	B.J.
	レモンパイ	L.P.
Sさん	アイスミルク	I.M.
	ストロベリーパイ	S.P.

そこで4人は，ジュースとケーキを自分の位置に動かさなくてはいけません。動かし方のルールは，①図のようなマス目を作り，となりのマスがあいているときには，そこへ動かすことができる。②まっすぐにひとつだけ飛びこえて動かすことができる。③テーブルの内側の部分は回転可能で，1ステップは90度とする。

以上の方法で，正しい位置に動かすには最低何ステップ必要でしょうか。自分の位置にあれば，ジュースとケーキの順番は関係ありません。ただし座席を変えるのはダメ。

PUZZLE BOX 3の正解

答　2枚

まず一般的に考えて，問題のアミダピースのそれぞれのピンに左端から0～6のピン番号を付けます。それぞれのピンから行先をたどると，

0→4，1→5，2→6，3→0，4→1，5→2，6→3――（1）

となっており，ブラックボックスのⒶから⑦に行くためには，

0→6

⇒(0→4)＋(4→1)＋(1→5)＋(5→2)＋(2→6)

の5枚のピースがいることになります。

しかしここで，意地悪な〝あ～みだ～くじ〟であるからには，もっと少なくて済むいやらしい方法があるのです。なぜなら，アミダピースは，左右反転，上下反転，水平回転によって4種のタイプのピースとなるのです。この場合，原形のP1とP4は同様に（1）の作用があり，反転したP2，P4は

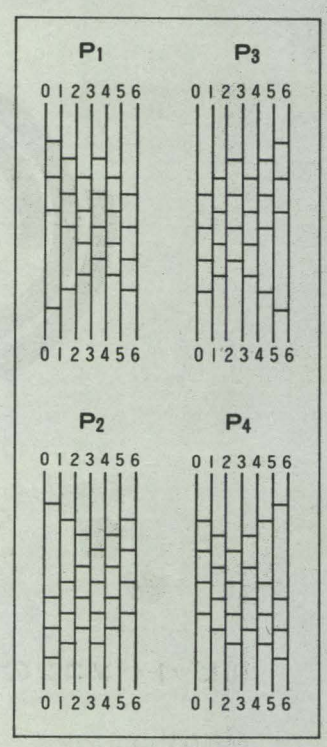
0→3，1→4，2→5，3→6，4→0，1，6→2 ――（2）

の作用がある。よってP2あるいはP4を使用すれば，（2）より

0→6

⇒(0→3)＋(3→6)

の2枚でできるわけなのです。ごめんなさ～い。



正解者

(愛知県)宇野高彦　(岩手県)湯澤健一　(佐賀県)村本一　(埼玉県)川端宏治様ほか6名の方に記念品を送らせていただきます。

P.S.　ところで〝MZ-5500プラモデル〟ってほんとーにあるんですか？……あったらほしい……。

応募方法

●官製ハガキに，答と住所・氏名・年齢を明記のうえ，
〒102 東京都千代田区四番町2-1
日本ソフトバンク　Oh！MZ　PUZZLE BOX 5係
までお送りください。

●**締め切り**　5月15日到着分　　●**正解発表**　7月号誌上

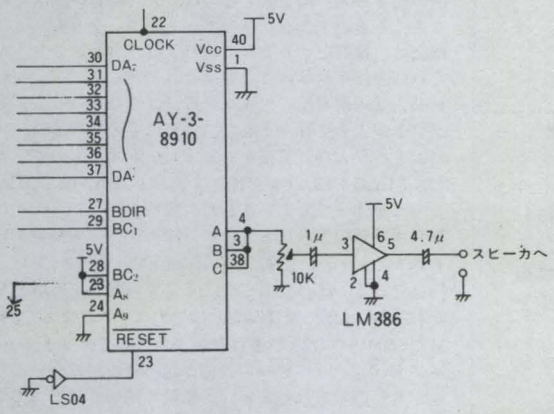
●正解者多数の場合，抽選で10名の方にOh！MZ特製記念品を差し上げます。

ごめんなさいのページ

本誌のバグに関するお問い合わせは
月～金曜日 16:00～18:00
☎03-265-5789

11月号 パラレルI/Oボードの製作

P.60 図2の回路図中に訂正がありました。



4月号 特集日本語ワードプロセッサ

P.43 中段12行目

MZ-5500のJWPの価格が85,000円となっていますが，89,800円（辞書ROMボード40,000円＋ソフト49,800円）の誤りです。お詫びして訂正いたします。

4月号 WORD WRITER

P.82 リスト中，以下の行を訂正します。

```
1290 LOCATE 32,16:PRINT"┌─ X─┐└──────┘│└────┘│└──────┘└─────┘"
1300 LOCATE 32,20:PRINT"┌─ Y─┐└──────┘│└────┘│└──────┘└─────┘"
1370 X1=(1+ABS(X+PX-1) MOD 6)*(A$=CHR$(19))+((X+PX)MOD 6-6)*(A$=CHR$(20))+(A$="■
")-(A$="■"):Y1=(A$="■")-(A$="■")
```

3月号 電子オルガン

P.56 リスト660～760行を以下のように訂正します。

```
660 '---PRINT---
670 PRINT[1,7]"┌──────────────────────────────┐"
680 PRINT[1,7]"└──────────MZ-700──────────────┘"
690 PRINT[1,7]"                                "
700 FOR J=1 TO 6
710 PRINT"     | ■ ■ | ■ ■ | ■ ■ ■ | ■ ■ | ■ |":NEXTJ
720 PRINT[1,7]"■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■■"
730 PRINT"     | | | | | | | | | | | | | | | |
740 PRINT"     |A|Z|X|C|V|B|N|M|,|.|/|?|)|=|*|
750 PRINT"     └─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┴─┘
760 PRINT"                    ■ ■ ■ ■ ■
```

3月号 パソコン千夜一夜

P.82 右段リスト

2070行の最後に 2120 を追加

3月号 PIRATE SHIP

P.106 リスト中

2320行の最後，GOSUB50 を GOSUB500

2440行の最後，GOSUB50 を GOSUB500

4月号 キューブプログラム

P.118 リスト130行と150行を以下のように訂正します。

```
130 COLOR7:CURSOR3,21:PRINT"WHAT DO YOU DO ?";
150 CURSOR3,23:PRINT"WHICH DO YOU MOVE ?";
```

SHARP MZ SOFTWARE INFOMATION

（シャープ株式会社MZソフトウェアインフォメーションNo.11より掲載）

●MZ-80B BASIC不具合い修正方法の件②

〔内容〕

①SB-6520において，SAVE/T，VERIFY時，SHIFT BREAKでメモリの内容が消える。また，MZ-80P4接続時，136桁以上のTAB設定に不具合いが判明。

②MZ-6620において，MZ-80P4接続時，136桁以上のTAB設定，および指数計算に不具合いが判明。

〔修正方法〕

AUTO RUNによるPOKE文修正の以下のプログラムにより，訂正してください。

①SB-6520 AUTO RUN PROGRAM

```
10 DEF KEY(1)=RUN ↵
20 DEF KEY(2)=LIST↵
30 DEF KEY(3)=CONSOLE
40 DEF KEY(4)=CONT↵
50 DEF KEY(5)=AUTO↵
60 DEF KEY(6)=CHR$(
70 DEF KEY(7)=DIR FD1↵
80 DEF KEY(8)=DIR FD2↵
90 DEF KEY(9)=DIR↵
100 DEF KEY(10)=LOAD ↵
110 REM
200 REM VERIFY,SAVE ｼｭｳｾｲ
210 POKE $4F5C,$11:POKE $4F5D,$51
220 REM
300 REM MZ-80BP4 TAB ｼｭｳｾｲ
310 POKE $534E,$A0:POKE $5352,$A0
320 REM
400 NEW
410 END
```

②SB-6620 AUTO RUN PROGRAM

```
10 DEF KEY(1)=RUN ↵
20 DEF KEY(2)=LIST↵
30 DEF KEY(3)=CONSOLE
40 DEF KEY(4)=CONT↵
50 DEF KEY(5)=AUTO↵
60 DEF KEY(6)=CHR$(
70 DEF KEY(7)=DIR FD1↵
80 DEF KEY(8)=DIR FD2↵
90 DEF KEY(9)=DIR↵
100 DEF KEY(10)=LOAD ↵
110 REM
120 REM
200 REM MZ-80BP4 TAB ｼｭｳｾｲ
210 POKE $50A9,$A0:POKE $50AD,$A0
220 REM
230 REM
300 REM ｼｽｳ ｹｲｻﾝ ﾌｸﾞｱｲ ｼｭｳｾｲ
310 POKE $373E,$CD
320 POKE $373F,$3A
330 POKE $3740,$5B
340 POKE $3741,$00
350 FOR I=23354 TO 23363
360 READ A:POKE I,A:NEXT I
370 DATA 197,205,194,46,193,62,32,203,126,201
380 NEW
390 END
```

編集室から

'84.6月号　5月18日(金)発売

●グラフィックツール
●Cコンパイラ入門

投稿をお待ちしています

小誌では①テーマ別、②ショート、③その他の3つに分けて、皆さんからの投稿を募集していますが、今回はテーマ別はお休みです。②③でご応募ください。掲載に当たっては、多少修正させていただくことがありますのでご了承ください。なお、投稿作品の中から特に優秀なもの1点に「月間特別賞」として、原稿料とは別に記念品を差し上げます。今回の賞品はソニーの新型ウォークマン(WM-20)です。

また、東京近郊にお住まいの、ゲームに強く、THE SOFTOUCHのテスト・ランを手伝っていただける18才以上の方を募集します。次の要領でお申し込みください。

①自己紹介文(400字程度) ②マイコン歴 ③お持ちのマシンと周辺機器 ④推薦するゲームソフト名とその理由 ⑤住所、勤務先、学校名と連絡電話番号を、封書にして、Oh! MZ編集室「テスト・ラン」係にお送りください。なお、お電話でのお問い合わせはお断りいたします。

定期購読のお知らせ

定期購読の申し込みをお受けしています。本誌が手に入りにくい地区にお住まいの方、毎月購読していただいている方、入手確実な定期購読をお勧めします。詳しくは本誌とじ込みの振替用紙をご覧ください。

バックナンバー在庫状況　'82.8、10、11、12 '83.8、9、10、12月以降の在庫がございます。お問い合わせは、出版営業(03-261-4095)宛お願いします。なお、バックナンバー常備店は143ページに掲載してあります。

海外送付ご希望の方へ

本誌の海外発送代理店、日本IPS(株)にお申し込みください。なお、購読料金は郵送方法、地域によって異なりますので、下記宛必ずお問い合わせください。

日本IPS株式会社　〒101 東京都千代田区神田小川町3-5 ☎03-291-2632(直)

広告目次

アコリット……184
アピコン……171
アマノ電器サービスセンター……176
ウスキパソコンセンター……172
エイト電気……165
SPS……181
久留米マイコンセンター……177
コスモス岡山……180
コンピュータII……170
コンピュータ・ビング……161
J&P……表3
実務教育研究所……107・119
シルバーボール……6
シャープ……1～5・表2・表4
スガヤ無線商会……173
ストラットフォードコンピューターセンター……8・9
タスクフォーツ高知……112
九十九電機……162
デービーソフト……174・175
東海クリエイト……7・163
トーカイプラザ……179
日経マグロウヒル……差し込み
ハドソン……10・11・166～169
パーソナルメディア……160
ピーシーエー……164
プロコン……12
BASIC HOUSE……182・183
マイクロラブ……178

SHIFT　BREAK

▶永い冬からやっと春と感じられる頃となった。学生生活の始まる人、職業人とならられる人、新しい生活の始まる人、も多いと思う。キーボードが嫌いな人も、これを機会に、練習を始めてみてはどうだろうか。もちろん、それ以上出来れば言うことなし。(H)

▶某大学の研究室のマイコンに、「メルコムごっこ」なるソフトが入っておりました。内容はその名の通り、メルコムという大型計算機をシュミレートするもので、実際に、FLAGやEDITが可能です。私も、BASICで、CP/Mごっこを作ってしまいました。根暗!!(G)

▶新入社員の入社にともない、編集室内の大移動が行なわれました。その際、ねずみ(MZ-5500のマウス)が行方不明になってしまったのです。そこで毎晩ねずみ取りをしかけておいた所、先日みごとにワナにかかりました。現在MZ-5500につないで逃げないようにしてあります。(S)

▶アドベンチャーゲームの紹介記事を安請け合いしたときから私の不幸は始まった。原稿の締切りは迫ってくるのに謎は解けない。この苦しみを丸5日間も続けてしまった。朝から晩までひとりでマイコンの前に座っている姿は思い出しただけでもゾッとする。(KO)

▶BYTEの最新号がやっと届いた。ひもじい生活である私は sea mail の一ヶ月遅れ。つまり2月号であるがこの特集 "Apple's Macintosh" には驚いた。68000 CPU、3.5インチマイクロフロッピー、マウス、ウィンドウ、etc。話によるとJWPを今秋サポートするらしい。(GTG)

▶私はバッグを左肩に掛けて歩くのを常としているのだが、先日なにげなく右肩に掛けてみた。すると、あな、恐しや、バランスが狂ってまっつぐ歩けないのである。以前から猫背だったから、これで前後左右に背骨がまがったことになる。私の体はどうなるのだろうか?(T)

▶梅が咲くと梅見で一杯、桜が満開になると花見で一杯。知り合い筋がやれ入学だ、合格だとお祝いで一杯と、いよいよめでたい季節になりました。5月になれば鰹もおいしくなるし、真にハルランマンであります。いやあ、実に楽しい今日このごろだあ。まいった、まいった。(R)

▶今月はまさにアドベンチャーシンドロームであった。残念ながら紹介しきれなかったものもあるが、正直に言うと、ペイントのトロさがわざわいしたものが多い。これらの作品には「私は、ドジでマヌケでノロマなカメです。」と反省し、努力してもらいたい。ジャンプ!(T)

▶もう3月も末だというのに寒い日が続き、春はどこから来るのかな?　などと思いつつ歩いていたら、一匹の巨大なカエルが私の眼に飛びこんできた。あまりに突然に春が来たので、思わず飛びのいてしまった。これでまだ確実に1mは春が遠くなった…かな。(K)

▶絵夢絶登玉手箱の連載がしばらく中断して皆様を不安がらせていますが、じつは編集部の特注で筆者の有田隆也君、友人のItti君と協力して、XIの解析にのめりこんでいるのです。来月号あたりからは第一線に復帰するでしょう。もうしばらく待ってね。(ja)

Oh!MZ 5月号

■1984年5月1日発行　定価480円　■発行人　孫　正義　■編集人　田鎖洋治郎
■発売元　(株)日本ソフトバンク
■出版部　〒102 東京都千代田区四番町2-1　☎03(261)4095　FAX 03(262)8397
■本　社　〒102 東京都千代田区九段南2-3-14　靖国九段南ビル ☎03(263)3690㈹
TELEX　東京 232-4614　JSBTY　FAX 03(263)3660
■大阪支店　〒542 大阪市南区難波千日前5-19　河原センタービル3F
☎ 06(644)0191㈹　FAX 06(644)0160
■印刷　凸版印刷株式会社

©1984 JAPAN SOFT BANK 雑誌 02179-5　本誌からの無断転載を禁じます。

●編集　三武るみ　前田　徹　菊川良子　安田千尋
●技術　松田辰夫　小林初雄　徳永　聡
●海外協力　Hisayoshi Mikami　Hong Liang Lu　Ian Allen　Ronald N.Billings
●協力　有田隆也　高野庸一　桑野雅彦　青木秀浩　西野文広　Itti Rittaporn　小笠原佳人　杉浦勇一　後藤貴行
●カメラ　浜崎　昭　杉山和美
●イラスト　永沢しげる　山田晴久
●レイアウト　森元　功　中島真子　渋井年雄　岡谷勝人　CAN ART　外山雅一　元木昌子

株式会社日本ソフトバンク発行の **Oh! シリーズ**

月刊
Oh!PC
5月号
480円
好評発売中！

特集：PC・ざ・ゲーム
ジャンル別ゲームをカラーで紹介
アメリカン・ドリーミング　APPLE, IBM PC, ATARI

◆続・泉のグラフィックコントローラ
◆PC-8801MKIIはこうかわった！
◆スーパーシステム88
◆8086アセンブリ言語講座(とじ込み付録・オペコード表)

月刊になった
Oh!FM
5月号
480円
好評発売中！

特集：日本語ワードプロセッサ徹底研究
ワープロソフト全比較／ソフト別試用感
ユーザの声／簡易ワープロ

▶OS-9ディスクエディタ
▶FM-7を解析する
▶レイトレーシングを用いた3次元CG
▶アドベンチャゲームの作り方
▶「FMモニタ」汎用ルーチンの解説

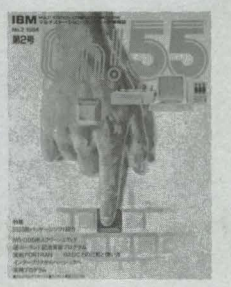
季刊
Oh!HC
第6号
480円
好評発売中！

特集：即戦アプリケーション集
学校成績処理／電子手帳／通貨変換／土木工事の見積
テニス・バドミントン成績集計

■ハンドヘルド3Dグラフィックスの世界
■6301アセンブラ
■ニューバージョン　QC-10 IIの全容

第7号　4月28日発売予定

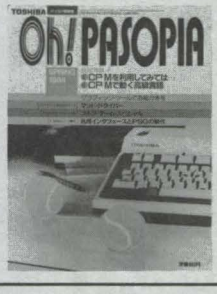
季刊
Oh!55
第2号
480円
好評発売中！



特集：5550用パッケージソフト

◉アプリケーションプログラム紹介 ビジネス グラフィック他
◉逆ポーランド記法実習プログラム
◉MS-DOS用スクリーンエディタ
◉マクロアセンブラ
◉ケーススタディ　5550導入オフィス
◉プログラム　バイオリズム相性率ほか

季刊
Oh!PASOPIA
第3号
480円
好評発売中！



特集：きみも言語博士になれる！ CP/Mを利用してみては…

⊙グラフィックツールでお絵かきを
⊙マッド・ドライバー(オリジナル3Dカーレース)
⊙汎用インタフェースとP.S.G.の製作
⊙活用しよう！　パソピア・データボックス
⊙パソピア7シンセサイザだ！

季刊
Oh!Hitbit
創刊号
480円
好評発売中！

特集：SMCグラフィックス
ソニー・グラフィックスエディタ／Computer Graphics
BASIC／タートル・グラフィックスに挑戦しよう
「SMCグラフィックスの世界」

★ニューメディアとHiTBiT
SMC-777の全容／CP/Mガイダンス／日本語ワープロ

★コンピュータ・グラフィックス最前線：JCGL
——「SF新世紀・レンズマン」

PERSONAL MEDIA CO

グラフ8

GRAPH 8

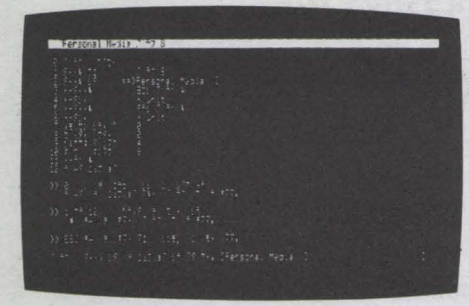
Copyright © 1983 by
Personal Media Corporation

MZ-7000シリーズ
プロッタプリンタ用
カセット版 ￥12,000

1. 折れ線グラフ
2. X－Yプロット（散布図）
3. 縦棒グラフ
4. 横棒グラフ
5. 円グラフ

PERSONAL MEDIA CORPORATION

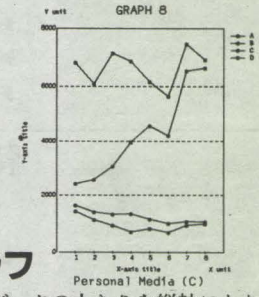
どなたにでも簡単に美しいグラフを作ることができます



現代のビジネスでは、毎日多量のデータを効率良く処理して行くことが必要です。データの速やかな把握、判断のためにはデータをグラフとして表わし視覚化するのが効率的です。ところが、手作業によりグラフの作成を行うには多くの労力と時間を要するため、従来はごく重要なデータの処理を行う場合にのみグラフ化が行われてきました。

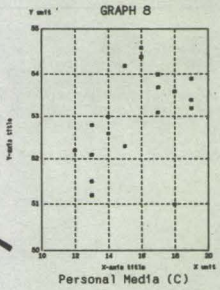
グラフ8は、パーソナルコンピュータからの簡単な質問に答え、表を埋めるような簡単な操作でデータを入力するだけで、棒グラフ、折れ線グラフなどのグラフがどなたにでも簡単に、美しく、そして速やかに作れるパッケージ・ソフトウェアです。しかも、入力したデータは保存し、後で再び利用することができますので、同じデータで表示形式を変えたグラフを作成するようなことも簡単に行うことができます。

1 折れ線グラフ



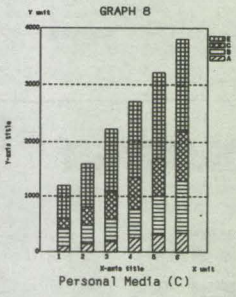
折れ線グラフはデータの大きさを縦軸にとり、データを順番に結んだグラフです。この折れ線グラフはデータの時間的な推移を見るのに適しています。

2 X－Yプロット（散布図）



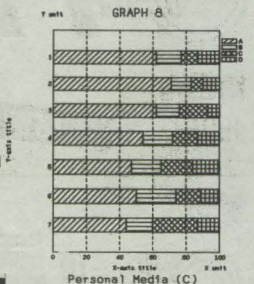
X－Yプロットは二つのデータの大きさを縦軸と横軸にとり、各サンプルを二つのデータの大きさを表わす線の交叉した位置の点として表わしたグラフです。このX－Yプロットは二つの性質間の関係（相関）を見るのに適しています。

3 縦棒グラフ



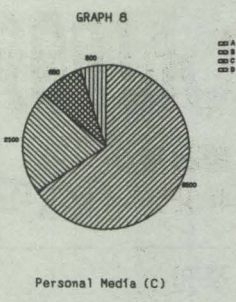
縦棒グラフはデータの大きさを縦棒の長さで表わすグラフです。このグラフはデータ間の大小関係を比較するのに適しています。また、一本の棒をいくつかの成分の大きさで区切ったいわゆる帯グラフも作成できます。このグラフは各サンプル内の構成比率を表現するのに適しています。

4 横棒グラフ



横棒グラフはデータの大きさを横棒の長さで表わすグラフです。このグラフはデータ間の大小関係を比較するのに適しています。また、一本の棒をいくつかの成分の大きさで区切ったいわゆる帯グラフも作成できます。このグラフは各サンプル内の構成比率を表現するのに適しています。

5 円グラフ



円グラフはデータの大きさを扇形の広がりの大きさとして表わし、全体で一つの円となるようにしたグラフです。この円グラフはある時点での構成比率を表示するのに適しています。

パーソナルメディア株式会社 〒141 東京都品川区西五反田8-1-11 ☎(03)490-8841

創刊 日経バイト
NIKKEI BYTE

4月23日、
創刊前・春号を発行

「BYTE」と全面提携して、ついに誕生！パソコンの技術情報誌

特典付き
予約・受付中、
お申込みは
いますぐ

世界で最も高く評価されている高級パソコン誌「BYTE」

パーソナル・コンピュータの創世期1975年に登場して以来、世界のパソコン界をリードしてきた「BYTE」。「日経バイト」は、この「BYTE」と全面提携して創刊。パソコンの最新技術、最新製品について、詳しく解説していきます。

パーソナル・コンピュータの最新技術、最新製品を詳しく解説

「BYTE」の精選情報を翻訳・掲載します。

「BYTE」の記事の中から、わが国のパソコン界にとって重要な情報を厳選し、こなれたわかりやすい文章で翻訳・掲載(「日経バイト」全誌面の約35%)するとともに、パーソナル・コンピュータ、エレクトロニクスの高度な知識を持つ専門記者の直接取材で、国内外の最新情報をお届けします。

パソコンの最新技術を詳しく解説します。

「技術的な解説」「体系だった情報」「パソコン界を幅広くカバーする情報」を提供。パソコンの先進的な使い方を進める際に必要な新情報を、あらゆる角度から深く掘り下げて解説します。

最新のハード／ソフトを徹底的に解剖します。

国内外のハード／ソフト、周辺機器の中から注目すべき新製品を取りあげ、独自の視点から徹底分析。ハードの回路設計手法、ソフトのシステム設計技法、その特徴、設計の巧拙まで踏込んで解説します。

特典：いま購読をお申込みの方へ、創刊までに特別編集版を3冊無料贈呈。

●創刊前・春号…4月23日(月)発行　●創刊前・夏号…6月25日(月)発行　●創刊前・秋号…8月20日(月)発行

＊裏面で創刊前・春号の詳しい記事内容をご紹介しています。

●10月1日創刊　●月刊(毎月1日発行)

●A4変型判、毎号約200頁

●個人の年間予約購読・ご自宅郵送制

購読料金：いま購読をお申込みになれば、特別編集版が3冊プラスされて非常にお得です。

★1年購読(12冊)……6,000円(1冊当り500円)

★3年購読(36冊)……12,000円(1冊当り333円)

発行：日経マグロウヒル社

書店ではお求めになれません。お申込みは、裏面のハガキでいますぐ

パソコンの技術情報

創刊 日経バイト
NIKKEI BYTE

購読お申込みはいますぐ!

創刊前・春、夏、秋号の3冊を無料贈呈

この創刊前・春、夏、秋号は、創刊までに購読のお申込みをいただいた方だけのために特別に発行するものです。いますぐお申込になり3冊ともお読みください。

強力な情報網で、最新のパソコン技術情報を提供。

「日経バイト」を発行する日経マグロウヒル社は、日本経済新聞社と、世界有数の情報企業、米国マグロウヒル社の共同出資で設立された、技術情報の総合出版社です。「日経バイト」の発行に当っては、日経マグロウヒル社の取材網をフルに活用するほか、日本経済新聞社、米国マグロウヒル社も全面協力します。

【購読料金】

- 1年購読(12冊)……6,000円
- 3年購読(36冊)……12,000円

＊料金お支払いは、今秋創刊号をお届けしてからで結構です。

発行:日経マグロウヒル社

〒101 東京都千代田区神田小川町1-1

読者サービス・センター ☎(03)380-3157

【創刊前・春号の主な内容】

【特集】マルチウインドウ

- ユーザー・インターフェイスを変革するマルチウインドウ
- Concurrent CP/Mマルチプログラミング/マルチウインドウOS

コンピュータとそれを操作する人間とのインターフェイスを革命的に変える新技術が、パーソナル・コンピュータの領域にも浸透し始めた。マルチウインドウとマウスを特徴とする新しいユーザー・インターフェイスの動向と技術面の特色を探る。

一般解説

- パソコン間データ通信ソフトと通信プロトコル
- マイクロフロッピーVSコンパクト・フロッピー・ディスク
- 32032、モジュール指向の32ビットcpu

パーソナル・コンピュータ関連の技術トピックを広範囲に集め、開発者あるいは記者が詳しく解説。

システム・レビュー

- PC-100:マウスとビットマップと日本語処理

話題の新製品を対象として、その使用感、仕様上の注目点や問題点などを、さまざまな角度から評価。

自作講座

- ADPCM方式を用いた明瞭度の高い音声合成ボードを自作する

市販の製品には見当たらなかったり、あっても性能に満足できないコンポーネントを、安価で簡単に入手できる部品を使って自作。

byteネットワーク:パソコン分野の新しい出来事をいち早くレポート。

フォト・レビュー:写真による話題の新製品解剖。

新製品:パソコン関連のシステム、周辺装置、ソフトなどを紹介。

マイクロバイト:最新のニュースを一目で理解できる形で速報。

クラブだより、カレンダー……ほか

郵便はがき

料金受取人払

168-□□

(受取人)
東京都杉並区浜田山4-5-5
杉並南郵便局私書箱35号
日経マグロウヒル社
読者サービス・センター
NBY係行

杉並南局承認 232

差出有効期間 昭和59年12月19日まで (切手不要)

日経バイト NIKKEI BYTE 特典付(創刊前・春・夏・秋号贈呈)・予約購読申込書

読者のプロフィールを正確に把握しニーズに応える雑誌づくりを進める資料にいたしますので全項目にご記入ください。

お名前(フリガナ) 年齢 歳

自宅電話 市外局番 局番 番号

ご自宅住所(フリガナ)
□□□-□□

お勤め先名

会社電話 市外局番 局番 番号

ご購読期間(いずれかに○印を) 1年(12冊)6,000円(1冊当り500円) 3年(36冊)12,000円(1冊当り333円)

購読料金お支払方法(いずれかに○印を) 1. 郵便局からの払い込み 2. 銀行預金口座自動引落し

業種(具体的にご記入ください) 職種(該当のものに○印を)
1. コンピュータ(ハード)技術者 2. コンピュータ(ソフト)技術者
3. エレクトロニクス技術者 4. その他の技術者 5. プログラマー/アナリスト
6. 専門職(弁護士、医師、会計士など) 7. 教職 8. 事務・営業 9. 学生
10. 経営・管理職 11. 商工サービス自営 12. その他

Oh! MZ 09-42-4602

書店ではお求めになれません。
お申込みは
このハガキでいますぐ

挑戦者求む。素人可！

麻雀狂

《麻 雀 狂》

麻雀狂は現在もっともリアルにシミュレートされた麻雀ゲームです。

ゲームの特長

※麻雀にレベル設定を可能にすることにより、飽る事なくプレイを楽しめます。

※麻雀ゲームが「わからない」「できない」とのユーザーには、初期画面のレベル設定時にアマチュアレベルを選択して下さい。ユーザーが切牌の判断が出来無い場合、コンピュータの判断により、切牌等を自動的におこないますので、初級者は適切な切牌の判断が可能です。

　他のレベルを設定し、ユーザーが思考時間内に切牌が出来ない場合、コンピュータにより無差別に切牌がされますのでユーザーは、敏速な判断が要求されます。(レベル0を除く)

※タイマーの停止も可能です。

※ゲーム進行による各指示（リーチ、ポン、カン、チー、ロン等）は、そのつど画面左下に指示されますのでマニュアルが不要です。

※麻雀狂では、他の麻雀ゲームには無い、多くの画面変化がもりこまれていますので、画面をごらんいただくだけでも楽しいと思います。また画面変化をする事により適度な思考時間を得ています。

※このプログラムは漢字ROMを必要とせず、漢字表示を可能にしています。

※リピート機能……同じ配牌によりゲームが可能。

※リプレイ機能……ゲームを中断し最初からゲームが開始されます。

《機 種》●シャープ／X-1シリーズ
●NEC　／PC-8801シリーズ
●富士通／FM-7

《価 格》●カセット／￥3,800
●5D　／￥5,800
●CD　／￥6,800

X1を愛する君

マシン語がわかる人もわからない人も、これだけは準備したいですね。

VINGMONITOR
テープ￥6,000

●BASICが不要なため大きなマシン語プログラムの作製が可能。
●アッセンブラ、ディスアッセンブラがこの1本で動作しますので作業効率が良い。
●バックアップ機能付。

㊟バックアップ機能は個人的な使用以外は使用しないで下さい。

お求めは、全国有名マイコンショップ・シャープエンジニアリング及び日本ソフトバンクをご利用下さい。お近くのお店で、手に入らない人は、プログラム名、カセット及びディスクの別、住所、氏名、電話番号を明記の上、現金書留でご注文下さい。(送料無料！)機種名もお忘れなく。

COMPUTERS' COMMUNICATION VING ビング
〒013 横手市八幡石町27
☎0182(33)4767

※当社プログラム等の質問に関してのお問合わせは木曜日を除くpm1:00～pm6:00迄にお願いします。

お申し込みは今すぐお電話で03-251-9911へ

ツクモゴールデンウイーク特別セール

4/18〜5/18

頭金ゼロ、毎月のお支払いゼロ
お支払いはボーナス2回払い——ツクモウルトラクレジット

目玉商品

1. MZ-2000（本体定価￥218,000）＋ツクモオリジナルソフト5本（￥15,000）
   ＝合計定価￥233,000➡特価¥89,800
2. CZ-800C（本体定価￥155,000）＋CZ-800D（14"カラーディスプレイテレビ¥113,000）＋CZ-8GR（グラフィックRAM￥32,000）
   ＝合計定価￥300,000➡特価¥169,800

※台数限定 品切れの際は御容赦下さい。

特注 ビジネスソフト業務 ついに開始!!

ビジネスオーダーメード
ソフト低価格大系を実現!!

- 今、貴方のほしいビジネスソフトはありますか？
- そのソフトは貴方のビジネスにあっていますか？
- 今、持っている貴方のビジネスソフトは100%活用していますか？

自分でソフトを作っているあなたは思う通りに作れますか、プログラムなど大変手間が、かかるものです。
そんなあなたと充分検討を重ね制作するスタッフをご紹介致します。

各社・各店・一般の方、是非九十九に御相談下さい
詳しくは☎251-9966

ツクモ〈イベント〉スケジュール

- ■4/21〜30 トレードセンター 商品先渡しセール（ニューセンター店）
- ■5/5〜6 じゃんけんセール（7号店） じゃんけんに勝ったらソフト2割引
- ■5/13 ニューセンター店にて 中古フェア実施!!
- ■5/15〜23 7号店2周年記念企画 ★1万円以上、3万円以上、10万円以上お買い上げの方にお楽しみプレゼントなど用意しておりますヨ！

ツクモトレードセンター 中古フェア5/13㊐

特価品一覧表
ニューセンター店
店頭だけの販売です

MZ-2000……¥74,800
MZ-1R01、02、03付
MZ-731……¥44,800
MZ-721……¥34,800
etc.

※上記はほんの一例です
この他たくさん用意しています

シャープ パソコンテレビX1

組み合わせシリーズ

X1C NEW

アクティブタイプ
CZ-801C 標準価格￥119,800

14型カラーディスプレイテレビ
CZ-801D 標準価格￥99,800
（マスクピッチ0.45mm）

X1D

プロフェッショナルタイプ
CZ-802C 標準価格￥198,000
（パーソナルコンピュータ＋キーボード）
14型カラーディスプレイテレビ
CZ-802D 標準価格￥128,000
（マスクピッチ0.4mm）

NEW

目玉商品

CZ-801C	本体	￥119,800
C14VC1	14"パソコンTVファインピッチ2000文字	￥93,000
ツクモオリジナルソフトテープ版3本		¥ 9,800
JOY-X	ジョイスティック	¥ 2,800

＝合計定価~~￥225,400~~➡特価¥159,800

目玉商品

CZ-801C	本体	￥119,800
CZ-140DS	14"カラーディスプレイテレビ	¥ 89,800
ツクモオリジナルソフトテープ版3本		¥ 9,800
JOY-X	ジョイスティック	¥ 2,800

＝合計定価~~￥222,200~~➡特価¥169,800

CZ-801C	本体	￥119,800
CZ-801D	14"カラーディスプレイテレビ	¥ 99,800
ツクモオリジナルソフトテープ版3本		¥ 9,800
JOY-X	ジョイスティック	¥ 2,800

＝合計定価~~￥232,200~~➡特価 お問い合わせ下さい

CZ-802C	本体	￥198,000
CZ-802D	14"カラーディスプレイテレビ	¥128,000
JOY-X	ジョイスティック	¥ 2,800
DISK版ゲームソフト1枚		¥ 4,800

＝合計定価~~￥339,600~~➡特価 お問い合わせ下さい

シャープ MZ-2200 組み合わせ

本体標準価格
¥128,000

●Aセット

MZ-2200	本体	￥128,000
IT02	データーレコーダー	¥ 19,800
NH-14DK	14"カラーモニター	¥ 69,800
カラーケーブル		¥ 1,800

＝合計定価~~￥219,400~~➡特価¥152,800

●Bセット

MZ-2200	本体	￥128,000
IT02	データーレコーダー	¥ 19,800
MB-27303	12"2000文字カラモニター	¥ 99,800

＝合計定価~~￥247,600~~➡特価¥162,800

ツクモトレードセンター大好評!!

マイコンを売りたい人、買いたい人の橋渡し

貴方がパソコンを売りたい・買いたいと思った時、ツクモトレードセンターにお問い合わせ下さい 当センターで動作状態が完動か整備かをチェックし、商品を充分査定の上、お客様の希望価格を設定してツクモトレードセンター中古品リストに登録致します

受付 トレードセンター
☎03-251-9977

受付 ニューセンター店
☎03-253-0986〜7

■査定基準
1 メーカー及び機種名 S No.セットバージョン
2 使用年月
3 動作状態（完動か整備か必要か）
4 マニュアル、ケーブル等付属品の有無
5 オリジナルタンホール及びパッキンの有無

売りたい人 直接品物をお持ち頂いても結構です その場で打ち合わせをし、トレートセンターお客様カートに内容等記入致します お電話でお申し込みになった場合もお客様カートを記入し、送付致しますので条件確認の上、サイン及び捺印して頂き、申し込み金￥2,000を同封の上、返送して下さるようお願い致します

買いたい人 委託者同様ニューセンター店又は電話で受付致します お客様カートを承認の上、記入・捺印して頂き、送付してもらいます

取引方法 確認されたお客様カートが着き次第、御連絡致します その時に商品を当センターに送付して下さい ツクモトレードセンター技術スタッフが動作状態等査定し、再度御連絡して希望価格を設定致します（この場合、販売価格の変更も有りますので御了承下さい） お預かりした商品は店頭又は電話にて購入希望者を受け付け致します 希望先着順にお電話で商品内容を出来るだけ細かく説明致します

■トレード成立後代金は……
委託者の条件で手数料（含む振り込み手数料）を差し引いた金額をすみやかに振り込み致します

だからツクモで買う……ツクモウルトラシステム

大特価クレジット
……魅力の価格がクレジットで

頭金ナシ月々3,000円クレジット

低金利クレジット
…便利なクレジット60回までOK！

全商品保証付
……製品はすべてメーカー保証付

電話クレジット
……申し込みは電話一本でOK

高額下取制度
…ツクモ全店にて高額下取り買取りOK！

日曜配達（札幌店除く）
……より楽しく日曜日

お支払いは2ヶ月後
……スキップクレジットで

★今、お申し込みになればお支払いはあなたのご希望に（但し6ヶ月以内）
★ツクモラクラククレジット
お問い合わせは☎251-9911まで

通信販売の受付は
03(251)9911へ
ご注文の際は、ご注文になりたい商品名をご記入の上、現金書留にてお願い致します 20万円以上は銀行振込みも取り扱っております お問い合わせ下さい

富士銀行 神田支店
普通口座 No.894047

即決 ツクモ全国クレジットOK!!
- 現金特別価格でツクモクレジットが利用できます 残金のみに金利がかかります
- 60回払いまで出来ます 但し、1回の支払い額は3,000円以上
- その場で、お持ち帰りできるクレジットもあります
- 印鑑、身分証明書（免許証など）が必要です

- 取扱いクレジットカード：日本信販、JCB、DC、UC、VISA、MC、CF（尚、クレジットカードご使用の場合は店頭員にご相談ください

■ご注文商品の中、未着の品物がございました時は、通販受付電話で納品書No.をお知らせ下さい。尚、商品の詳しい取り扱い等は、各販売店までお問い合わせ下さい。

■通信販売は 〒101 東京都千代田区神田郵便局私書箱135 九十九電機㈱Oh！MZ係

九十九電機 株式会社

ニューセンター店 〒101 東京都千代田区外神田1-16-10 ☎03(251)0987
名古屋店 3F 〒460 名古屋市中区大須3-30-86 ラシオセンターアメ横ビル内 ☎052(263)1681
5号店 〒101 東京都千代田区外神田3-1-14 ☎03(251)0531 2
7号店 〒101 東京都千代田区外神田1-9-9 ☎03(253)4199
札幌店 〒060 札幌市中央区南二条西3-15-1 さっしんビルB1 ☎011(241)2299
ビデオセンター店 〒060 札幌市中央区南二条西4 乙井ビルB1-1F ☎011(241)3999

■定休日 東京各店は毎週木曜日と第3水曜日、名古屋店は毎週月曜日 札幌店（マイコン・ビデオ店）は第3木曜日

★セットの場合は梱包送料￥2,000を加算して下さい。

MZ-5500 シリーズ 5″2D版

PC-9800シリーズ 8″・5″2DD・5″2D版

PC-8800シリーズ 5″2D版

PC-100シリーズ 5″2D版

日本語ワードプロセッサ ユーカラ

この機能、この使いやすさで、この価格。驚異のコストパフォーマンスを実現!!

各¥28,000

1 文節変換で入力効率を追求。

カナ・ローマ字入力による文節変換方式を採用。動詞の活用、送りがな、接頭語、接尾語もそのまま入力—漢字変換。だから、スピーディ!

2 多彩な機能も、メニュー画面でラクラク操作。

操作メニューが、いつも画面に表示されていますので、文書作成、編集、管理、印刷などの多様な機能も、ファンクションキーでスムーズに、簡単に操作できます。

3 長文作成も気になりません。

一頁あたりの行数が自由ですから、頁づけを気にせず、長文作成ができます。

4 作表・計算機能で、さらに有効活用。

カーソル移動で自由に罫線が引け、作表も思いのまま。さらに、四則演算、四捨五入などの計算機能を備えていますので、幅広い用途に対応できます。

5 豊富で多彩な辞書構成。

基本辞書の他に、固有名詞辞書、接頭語付辞書、短文(熟語)辞書、外字辞書など5種類の辞書構成。だから、求める漢字がスムーズに得られます。また、短文(熟語)、外字登録機能あり。

6 ユーカラどうしで文書ディスクの互換性があります。

ユーカラ・ファミリー内で、文書ディスクのサイズ・密度がまったく同じ場合互換性があります。NECのPC-8800用8インチ日本語ワードプロセッサ(PS88-1011-SF)も同様です。

アッ!という間にベストセラー

ユーカラは、NEC発売のベストセラー商品、PC-8800用日本語ワードプロセッサ(PS88-1011-SF ©東海クリエイト)と同機能ですから、信頼性は実証済み。

対応プリンタも豊富です。

CTC 東海クリエイト
Computer Technical Creation
〒108 東京都港区芝5-13-18 MTCビル6F ☎03-456-4615
宮崎営業所 〒880 宮崎市高千穂通1-6-35住友生命ビル7F ☎0985-29-1966

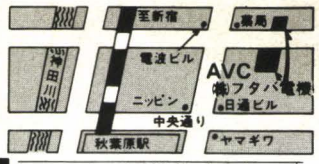
インフォメーションセンター
(専用)TEL 03-456-4610
●当社ソフトについてのお問合せは上記までお気軽にどうぞ

資料請求券
Oh / MZ 5Y

秋葉原価格でローンができます
電気の街秋葉原で20年の信用!!

AVC
AUDIO VIDEO COMPUTER

03(253)7661(代)



フタバ電機株式会社
〒101 東京都千代田区外神田3-2-3
コレクトコール(☎106)でお申込み下さい

もよりの電話へ
お申込み下さい。

札　幌 011-611-5104
新　潟 0252-75-4175
仙　台 0222-64-3704
名古屋 052-452-3271
大　阪 06-311-3931
広　島 082-295-6873
福　岡 092-481-2494

●富士通 FM-7

カラーグラフィック機能が充実、
入門から実務まで。

FM-7 ……￥126,000
▶均等払¥6,206×20回、¥4,351×30回

FM-7〈簡易言語付〉 …… ￥126,000
MB27303〈カラーモニタ〉 … ￥99,800
MB27501〈データレコーダ〉 ‥ ￥12,800
漢字ROM …… ￥35,000
GP250F〈プリンタ〉 …… ￥59,800
接続ケーブル …… ￥4,000
現金特価¥286,000　合計￥337,400
▶均等払¥14,620×24回、¥9,851×36回

IMBFDD2台、ワープロソフト(JWORD)、漢字ROM CP/M86、F-BASIC86付の強力ビジネスマシン。
FM-11BS(本体) … 新製品 定価￥398,000
現金特価¥340,000
▶均等払¥11,711×36回、¥9,208×48回
FM-11BS(本体) ……￥398,000
MB27315(12"カラーCRT) ……￥118,000
定価合計¥516,000
現金特価¥442,000
▶均等払¥15,224×36回、¥11,971×48回

IMBFDD1台、ワープロソフト(JWP-11)、漢字ROM、OS-9標準装備の未来指向マシーン。
FM-11AD2(本体) … 新製品 定価￥298,000
現金特価¥255,000
▶均等払¥8,783×36回、¥6,906×48回

MyOASYS2
ワープロ本体(6インチディスプレイ)
＋プリンタ(感熱式 ドット式 いずれかご指定下さい) ¥480,000
現金特価¥420,000
▶毎月¥9,467×36回、ボ¥30,000×6回
▶均等払¥14,467×36回

キヤノワードミニ
CM-5・定価 ……￥298,000 新製品
▶均等払¥9,231×36回

キヤノワード35
定価 …… ¥725,000
▶均等払¥17,600×48回

●NEC

PC-6601(3.5"マイクロFDD付)
¥143,000 現金特価¥125,000
▶均等払¥6,146×24回、¥4,306×36回

PC6001mkII

PC6001mkII 〈本体〉 ……￥84,800
PCDR321 〈データレコーダ〉 ‥￥19,800
PC60M43〈カラーディスプレイ〉 ‥￥68,500
PC60M75〈ディスプレイ置台〉 ‥￥5,500
PC6022 〈プリンタ〉 ……￥39,800
PC6094〈ケーブル〉 ……￥7,500
現金特価¥188,000　合計￥225,900
▶均等払¥9,243×24回　¥6,476×36回

ポケコンハンドヘルド

●カシオ
PB-700 〈本体〉
FA10〈プリンタ〉
CM-1〈マイクロカセット〉
……合計￥94,800
現金特価￥75,000
▶均等払¥6,938×12回、¥4,350×20回

●シャープ
PC1501 〈本体〉 合計￥114,600
CE150〈カラーグラフィックプリンタ〉 現特￥92,000
▶均等払¥5,336×20回、¥3,741×30回

●エプソン
HC-20 ‥￥138,600
H20MC 「ビジネスset」付 ‥￥25,800
現金特価¥130,000　合計￥164,400
▶均等払¥7,540×20回、¥5,287×30回

●キャノン X-07 ……￥69,800
▶均等払¥5,828×12回、¥3,654×20回
X-07R ……￥104,800
▶均等払¥5,452×20回、¥3,823×30回

入門コース 激安

FP-1100(本体) ……定価￥128,000
FP-1004(カラーモニタ)……定価￥98,000
FP-1082C(ケーブル) ……定価￥2,500
FP-1084CMC(ケーブル)……定価￥800
ゲームソフト1本サービス
定価合計¥229,300
現金超特価¥128,000
▶均等払¥6,293×24回、¥4,409×36回
FP-200(ハンドヘルド本体) 定価￥69,800
ライブラリー3冊 ……定価￥5,300
ゲームソフト2本 ……定価￥3,600
現金超特価¥54,800 定価合計¥78,700
▶均等払¥5,069×12回

●NEC PC-8801MKII

PC-8801MKIIモデル10(本体のみ)……
……￥168,000 現金特価¥145,000
▶均等払¥4,994×36回、¥3,927×48回
PC-8801MKIIモデル20(5"FDD1台付)……
……￥225,000 現金特価¥195,000
▶均等払¥6,717×36回、¥5,281×48回
PC-8801MKIIモデル30(5"FDD2台付)……
……￥275,000 現金特価¥235,000
▶均等払¥8,094×36回、¥6,365×48回

●シャープ
パソコンテレビX1シリーズ
一般テレビ放送受信可
CZ-801C S R〈本体〉
CZ-801D S R〈ディスプレイ〉 X1C 新製品
現金特価￥187,000 ……￥219,600
▶均等払¥6,441×36回、¥5,065×48回
CZ-802C S R〈本体〉
CZ-802D S R〈ディスプレイ〉 X1D 新製品
現金特価¥280,000 ……￥326,000
▶均等払¥9,644×36回、¥7,583×48回

ディスク

●NEC PC80S31 ￥168,000
PC9801・8801・8001・8001MKII・8201
▶均等払¥7,129×24回、¥4,994×36回

●エプソン TF10(PC用)‥￥129,800
▶均等払¥6,380×20回、¥4,473×30回
(FM用￥125,800もあります)

●工人舎 KD390PC KD290PC ……￥148,000
▶均等払¥7,714×20回、¥5,409×30回

●関東電子 K-305PC ……￥128,000
▶均等払¥6,264×20回、¥4,392×30回

プリンター

●エプソン
FP80 ……￥149,800
▶均等払¥7,366×20回、¥5,165×30回
FP-80K ……￥189,800
▶均等払¥9,338×20回、¥6,547×30回

●NEC
NM9300RO(漢字プリンタ)……￥253,000
現金特価¥225,000
▶均等払¥7,750×36回
NM9400RO(漢字プリンタ) ……￥310,000
▶¥9,300×36回 現金特価¥270,000
PCRP201 ……￥298,000
▶均等払¥12,538×24回、¥8,783×36回

●NEC

PC9801F

PC-9801E(N88ベーシック付)‥￥215,000
現金特価¥185,000 ▶¥6,372×36回
PC9801F-1〈16bit〉 ……￥328,000
CPU＋キーボード＋ミニフロッピー1台)
現金特価¥280,000
▶均等払 ￥9,644×36回、¥7,583×48回
PC9801F-2 ……￥398,000
(CPU＋キーボード、ミニFDD×2台)
現金特価¥340,000
▶均等払 ￥11,711×36回、¥9,208×48回

PC-100

PC-100model10 ……￥398,000
(CPU＋キーボード ミニFDD1台)
▶均等払 ￥11,711×36回、¥9,208×48回
現金特価￥340,000
PC-100model20 ……￥448,000
(CPU＋キーボード ミニFDD2台)
▶均等払 ￥13,089×36回 ¥10,292×48回
現金特価￥380,000
PC-100model30 ……￥558,000
(キーボード ミニFDD×2、CPU、
▶均等払¥16,361×36回 ¥12,865×48回
現金特価￥475,000
PCKD651(専用カラーモニタ)‥￥198,000
▶均等払¥5,787×36回、¥4,550×48回
現金特価￥168,000

●精工舎
GP500M(PC用新製品)……￥49,800
PCケーブル ……￥6,000
▶均等払¥2,799×18回 (PCケーブル付)
現金特価(PCケーブル付)¥44,000
GP550E〈漢字プリンタ〉 ……￥119,800
▶均等払¥5,568×20回、¥3,904×30回
CP700M〈カラープリンタ〉 ……￥99,800
▶均等払¥4,640×20回、¥3,253×30回

CRTディスプレイ

●シャープ
14M131C〈2000文字、8色〉 ……￥69,800
▶均等払¥3,248×20回、¥2,753×24回
14M132C 〈4050文字、16色〉…￥118,000
▶均等払¥5,510×20回、¥3,863×30回
14M142C〈高性能0.39mmピッチ〉
……￥99,800
▶均等払¥4,640×20回、¥3,253×30回
12M312C ……￥89,800
現特¥76,000▶均等払¥3,737×24回
CZ-140DS〈兼用モニタ〉 ……￥89,800
▶均等払¥4,176×20回、¥2,928×30回

●NEC
PC-8853N 〈超高解像度〉 ……￥168,000
▶均等払¥8,294×20回、¥5,815×30回
PCKD551 ……￥118,000
▶均等払¥5,916×20回、¥4,148×30回

●関東電子 K124(4050文字)¥115,000
▶均等払¥5,510×20回、¥3,863×30回

無料配送 全国どこでも無料。
製品先取り お申込み7日間位で届きます。
支払組自由 3~48回まで、ボーナス払も可、但し1回につき2,700円以上。
完全保証 即日対処のアフターケア万全。
日曜配送 共働きの方や、留守がちの方もお気軽にお申込み下さい。
全国クレジット どこからでもコレクトコール(☎106)でお申込み下さい。
頭金なし 手軽な電話クレジット。
お支払 は1回目が、2ヶ月後からです。
保証人なしクレジット 但し未成年、学生の方は保証人が必要です。

お忙しい方の為に
AM10時からPM11時まで受付 〈年中無休〉

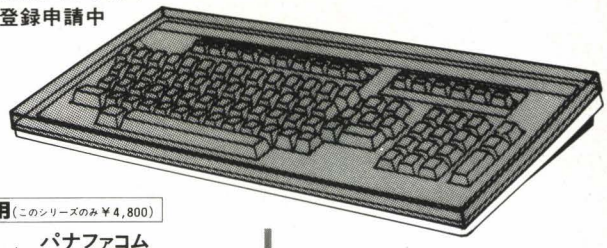
マイコン買ったらキーボードカバーも買おう！

キーボードカバー

(各)¥3,800
(OA用を除く)

実用新案登録申請中
意匠登録申請中

SOFTOP



シャープパソコン用

- ★X-1
- ★X-1D
- ★X-1C
- ★MZ-2200
- ★MZ-700

マイコンカバー(布製)
MZ-2000、MZ-80B用も取扱っています。
●カラー：エンジ・紺
¥3,000

NECパソコン用

- ★PC-9801
- ★PC-9801F・E
- ★PC-8801
- ★PC-8801MKII
- ★PC-8001MKII
- ★PC-6601
- ☆★PC-6001MKII

富士通パソコン用

- ★FM-7
- ★FM-8
- ▲FM-11

MSX用

- ☆★ナショナル CF-2000
- ★ソニー HIT BIT HB-55 HB-75
- 東芝 ★PASOPIA IQ HX-10D, 10S
- ☆★サンヨー WAVY-10
- ☆★富士通 FM-X

OA用(このシリーズのみ¥4,800)

- ▲NEC N-5200-05
- ▲富士通 9450

OA用(このシリーズのみ¥4,800)

- ▲パナファコム C280
- ▲ナショナル オペレート7000
- ▲IBM 5550
- ▲ソード M68

その他

- ★東芝 PASOPIA-7
- ☆セガ SC-3000
- ★アップルII, IIe
- ★日立 ベーシックマスター L3(2M,M5)

カラー	
	☆：透明
	★：ブラウン・スモーク
	▲：グレー・スモーク
材質：高級アクリル	

●ご注文の際は色の指定をしてください。

■：キーボードのまわりまでかぶさるタイプ
■：キーボードの一部がかぶさるタイプ
■：キーボードの上にのせるタイプ

※多量の特別注文は御相談ください。

別売品

宛名 マイコン用

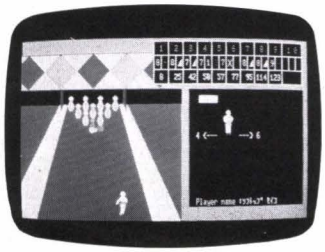
(型番 TC-10)

タックシール

●4インチ幅トラクタフィールド用紙
●1ケース・1ロール500枚入×2コ入
●シールサイズ(タテ)38mm×(ヨコ)86mm

¥3,200

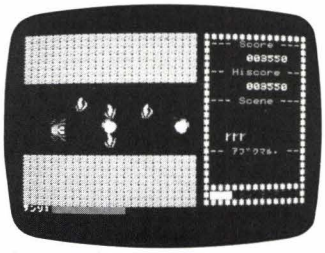
SOFTOPのゲームソフト



スーパーボウル

(X-1シリーズ) IPL. G-RM 型番 SX-003
(FM-7) 型番 SF-001 ¥3,500

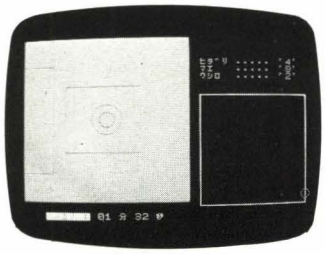
ハイ・ゲーム賞はだれの手に…



あぶく丸の冒険PART 2

(X-1シリーズ) IPL. G-RM
型番 SX-002 ¥3,500 J●Y

1面クリアーするごとに向きが 変わる。
裏返りしたあぶく丸を君は見れるか。



エイリアン・メイズ

(X-1シリーズ) IPL. G-RM
型番 SX-004 ¥3,300

迷路の中に3個のコイン。全部拾ったら出口へ急げ!! おっと、エイリアンがまちかまえてる。

ザ・ガードマン

(X-1シリーズ) IPL. G-RM
型番 SX-005 ¥3,500 J●Y

きみは9個のキー操作をどこまでこなせるかな。金塊を運ぶきみを待ちうける者は……

ブラックホール

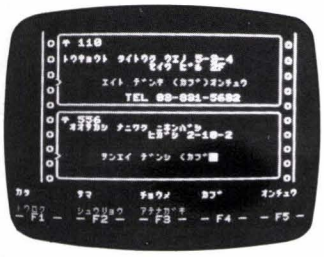
(X-1シリーズ) IPL. G-RM
型番 SX-006 ¥3,300 J●Y

きみのすごうでに、エイリアンもおもわずむしゃぶるい。ブラックホールの向うにあるものは。

あぶく丸の冒険PART 1

(X-1シリーズ) IPL. G-RM
型番 SX-001 ¥3,500 J●Y

次々とおそってくるタコのペトリゴンに立ち向かう、あぶく丸。落ちてくる岩のクッションを使ってペトリゴンをたおせるか。



宛名書き専用ソフト(住所録・電話帳)

ゆうびん屋さん

(X-1シリーズ) 型番 SX-501 ・(FM-7) 型番 SF-501
¥3,800

手書きの苦労が ふっとんだ!!

「ゆうびん屋さん」は機能を限定して、めんどうな操作や設定を除いた結果、はじめての方でも簡単に使っていただけるようになりました。

別売品

マイコン用 宛名 タックシール (型番 TC-10)

●4インチ幅トラクタフィールド用紙
●1ケース・1ロール500枚入×2コ入
●シールサイズ(タテ)38mm×(ヨコ)86mm

¥3,200

■お求めは……（商品送料はサービスいたします）
全国パソコンショップまたは直接右記へお申し込みください。
●現金書留でご注文の場合は商品名、住所、氏名、電話番号をはっきり書いてお送りください。
●銀行振込みでご注文の場合は商品名、住所、氏名、電話番号をハガキまたは電話でご連絡ください。

企画・製作
ソフトップ

発売元

エイト電気株式会社
〒110 東京都台東区上野5-3-4 ☎03-831-5632(代)
振込銀行：北陸銀行 上野支店 (普)No.4032110

三栄電子株式会社
〒556 大阪市浪速区日本橋東2-10-2 ☎06-643-3833(代)
振込銀行：三和銀行 恵美須支店 (普)No.189879

秘宝

「デゼニランド」にラブコールが到着！

三月磨臼ハンター続々誕生!!

デゼニかわら版

○青木英樹さん　宮城県　14歳

こんにちは、ハドソンさん。デゼニランド買いましたよ。むずかしいですね。最初に入場するのになんと20分もかかってしまいましたよ。ところで、今ボクはあるところで非常に困っております。そこでちょっとヒント（よーするにキーワードを教えてほしいんよ）をもらえないかな〜〜と思ったら、マニュアルにそーゆーことは教えられないと、シッカリ書いてあるではありませんか。エーッ！　ハドソンってそんなことするの。冷たいですねえ。それはないと思いますよ。このゲームは、ハドソンさんが自信満々に出したのだから、それはそー簡単に解けないでしょう。だから、それだったらある場所の抜け方くらい教えてほしいワケよ。ダメ！　あのさー、ホンのちょっぴりだけでもいいんだよね。

それでも、やっぱりダメ!?

○阿蘇実さん 新潟県　21歳

やっと冬が終わろうとしてますね。ハドソンの方々も心ウキウキ、ソフトもりもり、頭はグチャグチャで雪どけのようでしょうね。ムヒッ！　ところでネー、「デゼニランドめでたく終了しましたよ。えらく感動したところも、何だか変に感じた点もいろいろありました。ではゲームをふり返えりつつ書きますー。何せ一番古いX-1ですので、ゲームは当然カセット。ではPOWER ON！まず超高速ペイントにびっくり。タイトルはまあまあ。初めての時はいきなりバスにのっちゃった。えーとですネ、いきなり不満を述べますとですね。全般的にフラグの立て方が甘い！　それとホラマンションの絵、特に墓地の絵はもう少していねいに仕上げて、もう少し不気味に（オバケや木をていねいに）仕上げてほしかった。しかし、何といってもウメ下館！　これはヨカッタ。なんともナンセンスで！

○大川朋之さん 埼玉県　20歳

私は、めったに市販ソフトなど買いません。しかしたまたまデゼニランドを買って、最初はバカにして「何か、ダサイ名前だな」と思ってましたが、これがスゴイ。ビックリしました。ところで、私は今コウモリの画面で1ヵ月も悩んでます。もうお手上げ！コウモリ取れない。畜生、教えてくれ〜〜！

○大江昌明さん 岐阜県

ぼくはデゼニランドを買ってからまだ一ヵ月で、まだ「ホラマンション」のピアノに泣く中学三年生です。竹部さんの頭の構造はいったいどうなっているのですか。いままでぼくにかかって一ヵ月以上悩ませたゲームはありません。それだけに今燃えています。

ヤルゾ!!

○手嶋知己さん　宮城県　16歳

なかなかオモシロかった。最初のパッケージの印象は、とてもきれいで、4,800円にしては安いと思った。ストーリーもまとまっていたし、絵を描く速さにはビックリ!!　ディズニーランドを元にして作ったそうだが、ヘリコプターが上空を飛んだ時の地上の風景が、本物のディズニーランドのような気ができた。この手のアドベンチャーゲームがFM-7にもできるのか、とものすごく感動しましたよ。和英辞典をわざわざ買ってきて、このアドベンチャーゲームをやったが、単語の勉強にもなってとてもよかった。デゼニランドを作った中本さん、竹部さんに感謝します。おもしろいソフトを作ってくれて、どうもありがとう。がんばって下さい。

○岡田光広さん　東京都　16歳

なんとあの「デゼニランド」をとうとう終わらせてしもうた。1ヵ月もかかったんやど〜〜！　もう最初からひっかかったもんネ。なんたって○○してもムダだよなんて書いてあったにもかかわらず、○○して○○を拾うなんてさ！　むむっときてしまいますよ。そのあとの○○なんて、まさかあれを持って次に行くなんぞ全く気づかなかったぞ。さすがやね。次の第2パビリオンはそうそうに見えて、そうではないぞ。たった1箇所のキーワード探すのに1日かかったのもあったんやど。そんなワケで、まあなんとか最後までできたんやが、しかしナカモトシンイチタケベタカシは、ひねくれもんじゃ！

○山口光一さん　群馬県　24歳

先日めでたく「デゼニランド」が解けましたので報告します。まずペイントの速さにはビックリしました。中間色が原色と同じ速さで、またはそれ以上の速さで塗られていくからです。PC-8801と同じようにジワーと拡がっていくんですね。この感じはボクは大好き。それがFM-7で可能とは！

ボクにも言わせて……

●マジでちょうど六日と半日で終わったぞー。どうだスゴイだろ。長所も欠点もあるが、何だかんだと言いながら、日本では最高級。佐野賢司さん 静岡県

●第一作目としては内容も濃く、ペイントも速く、非常に興奮した。英語入力のアドベンチャーは初めてだが、何の抵抗もなくスムーズだったのは完成度の高さだろう。次回作が楽しみ。篠塚愛一郎さん　東京

●画面がきれいで絵がシンプルでオモシロイ。中間色が速くてとても良い。最高に難しく面白かった。本田祐司さん　愛知県

デゼニランドご入場の皆様、誠にありがとうございます。

このデゼニランドはアドベンチャーとしてはかなり長いものなので、何人ぐらいの人が最後まで行くかなと思っていたところ、たくさんの人から「終ったゾー」と手紙をもらい、作者として、とてもうれしく思います。

デゼニランドは、なんと言っても取材活動が成功の要因ではないかと思い、では、次回作もその線で……と考えたりしています。デゼニランドの次……ウン！次はデゼニワールドでいってみようか？そう、アメリカのマイアミにXXワールドというのがあるではないか！リアリティを追求するならばず現地取材をせにゃならん。こう思い、早急、我社の大蔵省、別名、鬼の経理課を拝み倒して取材のGOサインを得るべく、日夜意見調整に悪戦苦闘！「アメリカに行ってもいいかなアー」と叫べども「行ってもいいともー」は、なかなか聞えてこない。さて、どうなるか？　もしかすると、これもアドベンチャーなのかと思い死ぬほど笑ってしまった。そんなこんなで、デゼニワールドは、デゼニランド以上にパワーをかけるゾ！というところで、乞う御期待！　作者（竹・中コンビ）

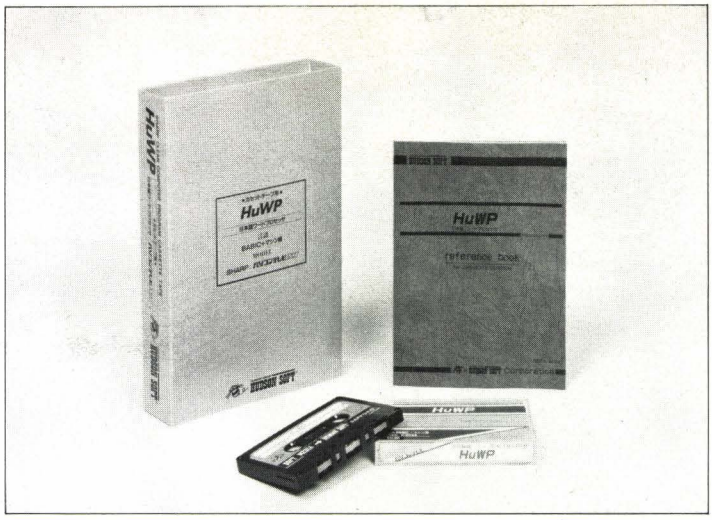
HUDSON GROUP
HUDSON SOFT®

HuWP

日本語ワードプロセッサ

簡単に使え，豊富な機能

SHARP
パソコンテレビX1，MZ-2000,MZ-2200,MZ-80B



HuWPの概要

このワールドプロセッサーソフトはパソコン本体と純正プリンターに漢字ＲＯＭの組み合せで使用します。使用可能な文字は、漢字ＲＯＭに含まれているＪＩＳ第１水準の2965字及び、片かな、ひらがな、英数字、一部の記号などは直接キーボードから、又漢字はかな漢字変換方式で簡単に入力することができます。この漢字かな変換は従来のテープベースのものでは非常に困難でしたが、かな漢字変換ＲＯＭを使用することにより、本体のメモリー空間を使用することなく約6400字の読み方辞書を持つことができました。また、ＲＯＭベースですので変換は大変高速に行えますし、もちろん漢字パターンの表示も漢字ＲＯＭのおかげで高速に表示します。

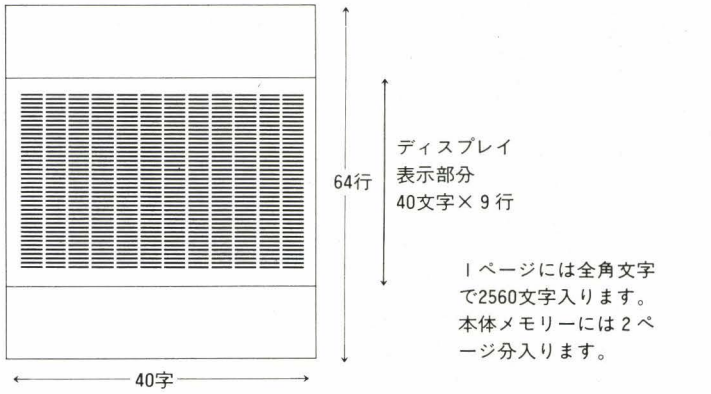
文章の編集機能も強力でコントロールコードを用いることによって、文章のコピー、移動、インサート(挿入)、デリート(削除)などが自由自在に行えます。また、文字のサイズは倍角、全角、半角が自由に選べアンダーラインも引くことが出来ます。作成した文書はカセットテープに記録または再生することが出来、それに要する時間は１ページ当り１分以内と高速になっています。

以上のようにこのワードプロセッサソフトHuWPは、フロッピーディスクを使用しない物の中では豊富な機能を持ち、フロッピーを使用しないことにより、トータルコストも大変安価になっています。

HuWPの特長

1）カナ漢字変換ＲＯＭの使用で漢字の入力が大変容易になった。
2）１ページには全角文字2,560文字を収容でき、本体メモリー内に２ページ記憶できる。
3）カセットベースのワードプロセッサでありながら、高機能である。
4）画面上に不要なマークを使用せず、印字される状態のまま表示になる。(倍角、半角なども画面上に表示される。)
5）用紙のサイズに合せて、色々な印字状態が設定できる。
6）文章全体のレイアウトを表示することができ、レイアウト表示中でも、通常の入力修正が出来る。
7）文章を１ページ、１分以内に記録再生できる。
8）カナ変換の時、読み方の前後をワンタッチで検索してくれる。
9）親切なHELP機能がある。
10フルマシン語でキータッチが良い。(X1は一部BASICで記述)

HuWPのページ構成



HuWPを使用する際必要なシステム

	パソコンTVX1	MZ-2200&MZ-2000
本　体	CZ-800C	MZ-2200 or MZ-2000
プリンター	CZ-800P	MZ-80P6 or 新MZ-80BP5 EPSON MP-80 TYPE3, 100 TYPE3 RP-80, FP-80はPIO-3050typeMP06 を接続すれば使用可
漢字ROMボード	CZ-8KR	PIO-3055
グラフィックRAM	CZ-8GR	MZ-1R01＋MZ-1R02(MZ-2000の場合) MZ-2200の場合は内蔵済

HuWPの実行

X1の場合

1）電源を入れ、本体附属のBASICテープをまずLOADして下さい。
2）BASICのLOAD終了後、本WPソフトをカセットにセットし、"RUN"して下さい。
3）マシン語部分を読み込みプログラムはAUTO STARTして初期画面になります。

MZ-2200&MZ-2000の場合

1）電源を入れ、本WPソフトをカセットにセットしてLOADさせて下さい。
なおBASICテープは必要ありません。
2）ソフトテープ読込み終了後、AUTOSTARTします。

MODEL：SHARP パソコンテレビX1	SERIAL X-3052-G ￥24,800（変換ROM付）
MODEL：SHARP MZ-2000, MZ-2200	SERIAL WB-3500A￥24,800（変換ROM付） WB-3500B￥19,800（変換ROM無）
MODEL：SHARP MZ-80B	SERIAL W-3500A ￥24,800（変換ROM付） W-3500B ￥19,800（変換ROM無）

HUDSON GROUP
HUDSON SOFT®
ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II201 PHONE:011-821-1538
ハドソン仙台/〒980 宮城県仙台市宮町1丁目4番28号 PHONE:0222-65-7031
ハドソン金沢/〒920 石川県金沢市本町2丁目1番28号 PHONE:0762-23-1263
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソン沖縄/〒901 21 沖縄県浦添市大平8番10号 PHONE：0988(79)3361

(4人マージャン)

ジャン狂

完全マシン語

徹夜にご用心！

これまでのソフトでは味わえなかった興奮。
"ジャン狂"は、スリリングなおとなのゲームだ。

IBM5550
5'FD ¥9,800
4月5日発売

1. Key操作が簡単
   カーソルの移動だけで拾牌処理(5つのKeyでリーチ、ツモ、カン、ポンなど全て可能)
2. 現在の持点が表示
   トップを狙うには相手の点数を常に意識していて下さい。
3. あがり点数の表示
   点数表示が非常に詳しいので役を覚えるのにぴったりです。
4. まち牌自動停止機能
   複雑なまち、例えばメンチンで頭を悩ますことなどなくなりました。
5. オープンモードも可能
   コンピュータの手作りを見ながら対局、あなたも負け組とはサヨナラです。

適応機種

X1-D X1シリーズ	3″FD T	¥6,800 ¥4,000
PC-9801/F	5″FD 8″FD	¥6,800 ¥7,800
PC-8801/MKII	5″FD T	¥6,800 ¥4,000
PC-6000シリーズ	T	¥4,000
FM-7	T 5″FD	¥4,000 ¥6,800

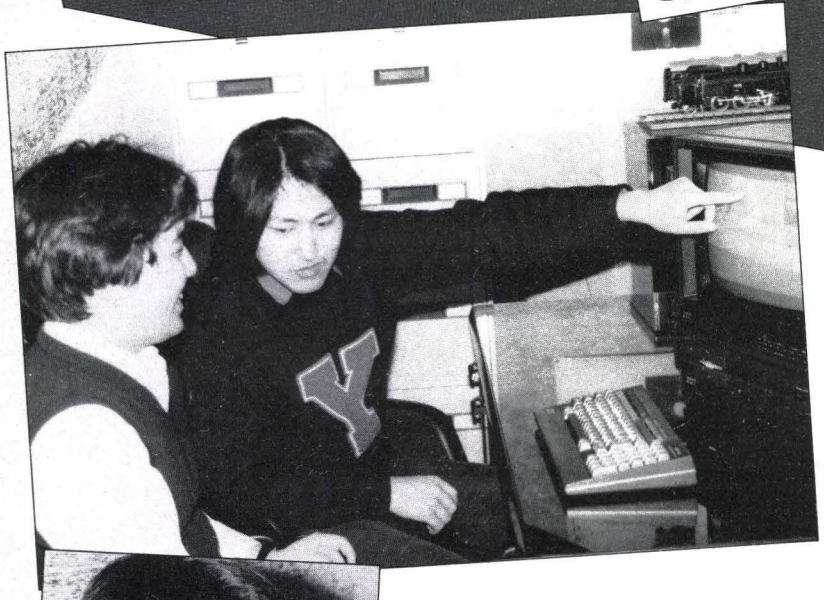
HUDSON GROUP
HUDSON SOFT®

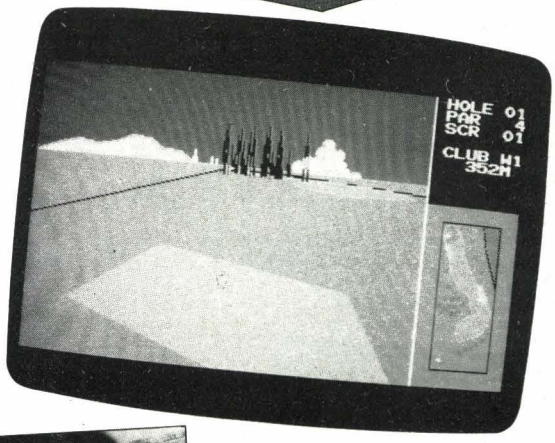
本社・ハドソン札幌／〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II201 PHONE:011-821-1538 営業所・仙台・金沢・東京・大阪・沖縄

ゴルフ狂 おもしろ体験記

Hudson soft

君はシングル？バーディーを
目指してティーショット！



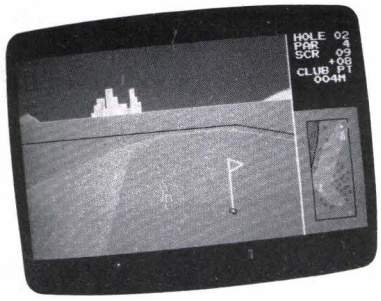


ゴルフ狂の作者 田中裕二

サア！ここはハドソン・カントリー・クラブ。むこうに見えるはテイネ山？只今から、本格的３次元ゴルフシミュレーションゲームの始まり始まり。プレーはアニメゴルファーのＴ選手。なあんて、実際に画面上でゴルフプレーが展開される。こんなゲーム、今まで見たことある？　サスガコンパクトフロッピー版。

この本モノそっくりのゴルフゲームにチャレンジするは、パソコンに関してはちとうるさい小久保秀次朗君高一。

小久保君「画面の中、むこうにホールのピンが見える。ハイ、ゴルファーのステップ位置はこれでよし。この角度ならピッタリ。打球の強さはと……。ゴルファーが１回だけ素振りをするから、２回目のスイングのときにテイクバック（ホラ、勢いをつけるために、クラブを振りかぶるアレだよ）のタイミングを見はからって、エイ！　と、スペースバーをたたくと、打球はアレ？　ＯＢだ……。エ？　ティーショットはアイアンで打つとフックがかかり易くなるんだって？上空の風にあおられちゃったのか、ゴルフなんて初めてだからなあ。」



おもしろ体験者
小久保秀次郎君(高一)

さて、小久保君、ウッドでやり直して、途中ラフに入れながらも、何とか池越えに成功。

ここでもう一度あたりの状況を見ると……ワンタッチで画面は広角に。広い視野をカバーできるんだ。ア！右横にバンカーがあるぞ。そこで、左に寄せて……と、アレー！左にもバンカー。ハア！バンカーはサンドエッジで強く出すんだって？　やっとグリーンにノったけど、ピンまで遠いなあ。いつになったらこのホール……。結局、パー４のところ、倍の８打たたいちゃった。でも確かにホントにゴルフをやってるような感じも。こりゃ慣れてきたらバーディーもホールインワンも夢じゃないぞ。」（私は全ホール回って７アンダーでした。作者）

感想「このやり方なら、マイコンボーリング、マイコンテニス、マイコン……何だって楽しくなると思うなあ！」

適応機種

PC-9801	5″FD	¥6,800	FM-7	T	¥4,000
PC-9801F	5″FD	¥6,800	FM-7	5″FD	¥6,800
PC-8801/mkII(共用)	5″FD	¥6,800	X1-D	3″FD	¥6,800
PC-8801/mkII(共用)	T	¥4,000	X1-C	T	¥4,000

HUDSON GROUP
HUDSON SOFT ®

ハドソン札幌/〒062 札幌市豊平区平岸3条5丁目4番17号 コロナード平岸II 201 PHONE:011-821-1538
ハドソン仙台/〒980 宮城県仙台市宮町1丁目4番28号 PHONE:0222-65-7031
ハドソン金沢/〒920 石川県金沢市本町2丁目1番28号 PHONE:0762-23-1263
ハドソン東京/〒102 東京都千代田区麹町4丁目7番5号 麹町ロイヤルビル2F PHONE:03-234-4996
ハドソン大阪/〒542 大阪市南区南船場4丁目2番18号 佐野屋橋ビル5F PHONE:06-251-1945
ハドソン沖縄/〒901-21 沖縄県浦添市大平８番10号 PHONE：0988(79)3361

出た〜っ!

全国有名書店・パソコンショップでお求めください。

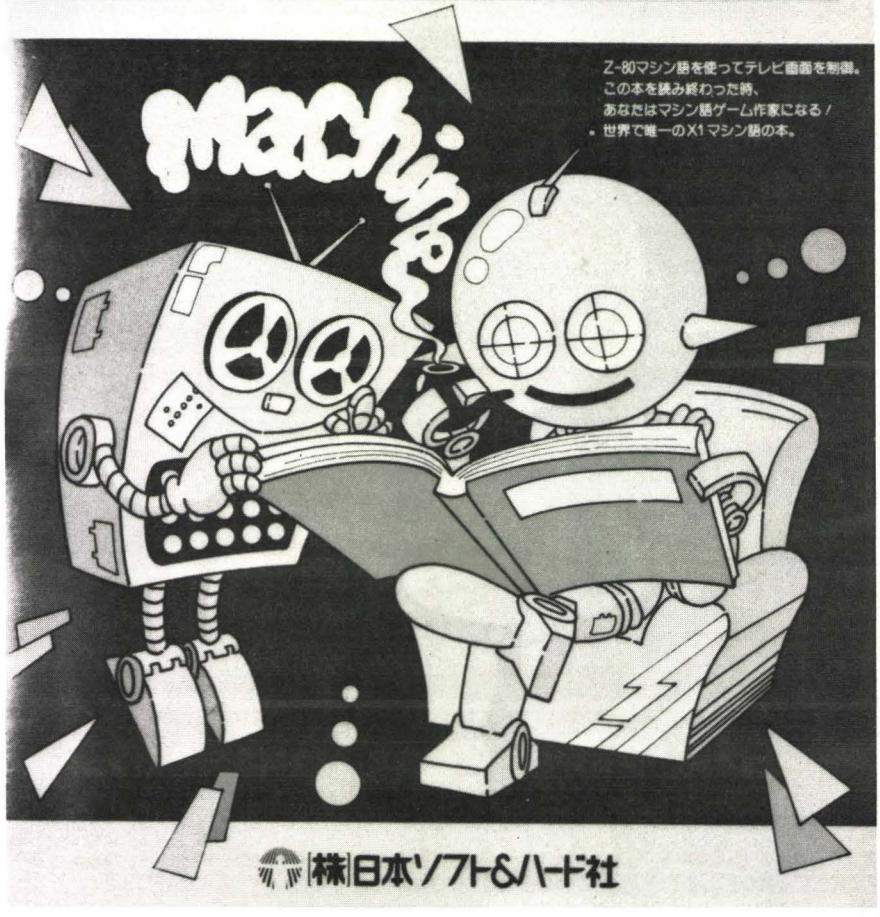
X1ユーザー待望の1冊!

X1/X1C/D完全対応!!

誰にでもわかる X1 マシン語入門

X1/X1C/X1D対応!

清水保弘 著



B5判/270ページ
定価2,800円(〒300円)

X1ユーザーの皆さん、お待たせしました。これでX1のマシン語はバッチリです。「マシン語なんて夢のまた夢さ」なんて嘆いてた人もどうぞ実物を手に取ってください。これからBASICをやる人も売り切れないうちに手に入れといてください。

近日発売予定のX1関連書籍

「X1マシン語・中級篇」

7月発行予定

「X1マシン語入門」でひと通り勉強したあとはこの本でさらにレベルアップ。これで君はX1の全てを知る

「X1ゲームライブラリー」

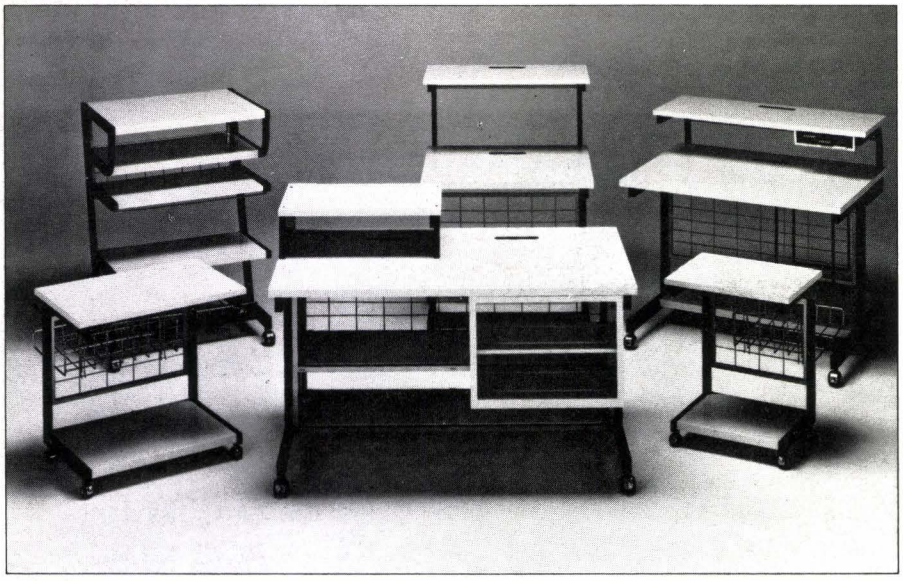
5月発行予定

X1にしかない機能をフルに引き出したプログラム集。ゲームを楽しんでいるうちにBASICもマスター。

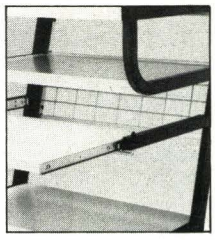
コンピュータ・イレブンの 株日本ソフト&ハード社

出版部／03(209)2585 高田馬場本店／03(209)7376 新宿店／03(342)4821 横浜店／045(312)4611
名古屋店／052(451)7371 大阪梅田店／06(346)1552 神戸三宮店／078(332)3961

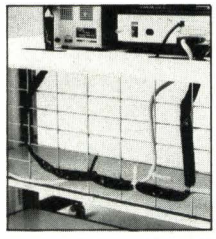
各社パソコンにジャストフィット!!



アピコンのパソコンデスクシリーズは各社のパソコンに適合したタイプがお選らびいただけます
詳しくはカタログをご請求下さい。

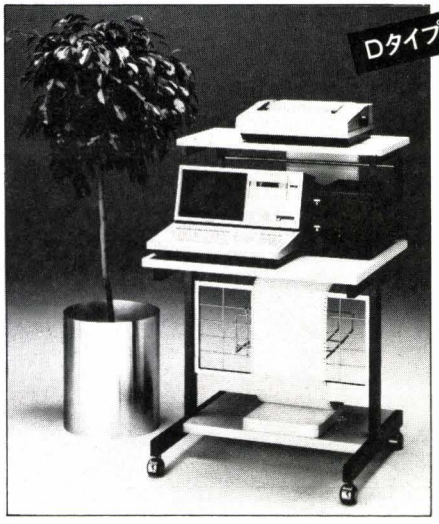


場所を取らないスライドテーブルは両サイド・ベアリング式のスムーズな出し入れでスペースを有効に利用いただけます。
(Aタイプに採用)



背面はワイヤーネット式で、コード類の配線に有効です。またオプションのペーパストレイジのハンガーとして使えます。
(各タイプに採用)

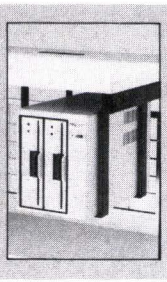
間口660×奥行500×高さ950㎜
価格構成
システム価格 ……￥45,000



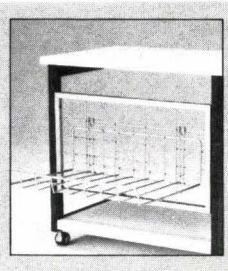
間口700×奥行600×高さ680㎜(＋335㎜)
価格構成
本体CS-30……￥25,000
ペーパストレイジPO-10……￥ 3,500
オーバートップデスクMO-40……￥11,000
システム価格……￥39,500

間口900×奥行700×高さ680㎜(＋185㎜)
価格構成
本体CS-10……￥33,000
オーバトップMO-10……￥12,000
デイスクドライブハンガーFO-20…￥ 3,000
フロッピートレーFO-30……￥ 3,000
システム価格 ……￥51,000

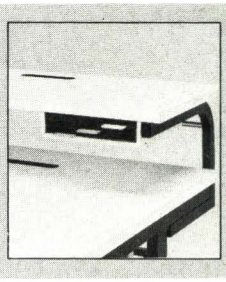
オプション



デイスクドライブハンガー
タテ型8インチ・5インチの2ドライブ対応型です。ネジ4本でカンタンに取り付けられます。
FO-20 ￥3,000



ペーパストレイジ
プリンターより打ち出された用紙を受けるための物です。
間口420×奥行320㎜
PO-10 ￥3,500



オーバートップ
トップ下有効165㎜
間口900×奥行350㎜
MO-10 ￥12,000

フロッピートレー
3段トレー式
8インチデイスク対応
FO-30 ￥3,000

商品は簡単な組立式で短時間で組立られます。〔送料に付きましては品種別にお問合せ下さい。〕 実用新案登録申請中

マイコンショップ・事務機店にて販売中です。

販売店募集中

APICON

アピコン
堺市築港浜寺西町7番地-4
TEL.——0722(68)3399
電話でのご注文も受け賜ります。
製造元 日本エレガント株式会社

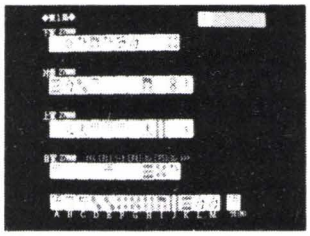
カタログ請求券
oh! MZ
4月

★新情報化時代への提案★

朗報

PC-6001MK-Ⅱ「本格的四人麻雀　雀鬼」のバージョンアップ（マシン語）盤が完成しました。①ゲーム進行のスピードアップ②実戦的判断等。尚旧バージョンお持ちの方は旧雀鬼テープと送料￥500をお送り下さい交換します。

ウスキオリジナルゲームソフト

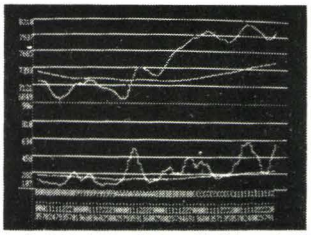


●本格的四人麻雀 雀　鬼

- ＭＺ5500　ＰＣ9800Ｆ
- ＦＤ　ＲＡＭ256Ｋ
- 漢字ＲＯＭ

￥6,800

他人にめいわくかけずに、１人でコンピュータ（３人分）相手に四人マージャン楽しいですね。又思考プログラムが追加されました。他にMZ2000、MZ2200、X1 ￥4,000、MZ80B、PC8001mKⅡ、FM7、PC6001mKⅡ、MZ-700、￥3,000

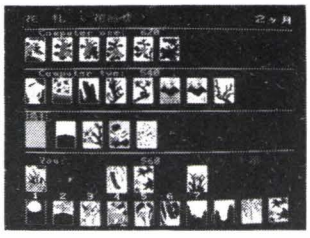


●株価チャートディスプレー

- ＭＺ5500　ＰＣ9800Ｆ
- ＦＤ　ＲＡＭ256Ｋ

￥28,500

1.ＤＡＴＡは、常に一番新しい３年分を記憶　2.移動平均線は任意の２種類を描くことができる　3.一画面は 142日分の日足を描く（約半年分）　4.色によって、ローソク足、出来高、引値足、移動平均線を描く。
他にＭＺ80Ｂ、2200、2000、Ｘ１、ＦＭ７、ＰＣ8001ｍＫⅡ￥ 3,500もあります。

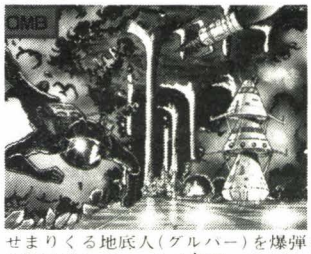


●花札（花鳥風月）

- オイチョカブ、花合わせ
- ＭＺ5500　ＰＣ9800Ｆ
- ＦＤ　ＲＡＭ256Ｋ
- 漢字ＲＯＭ

￥6,800

お金をかけずに１人でもまた、友達といっしょに仲よくやれる花札（オイチョカブ、花合せ）セットです。
他に花合せ　ＭＺ2000、2200、Ｘ１、￥4,000　Ｘ１Ｄ￥4,800

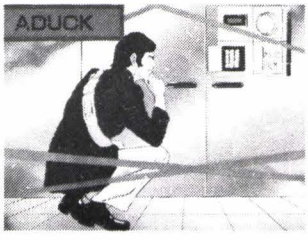


●ＤＯＷＮ ＢＯＭＢ

- ＭＺ 700
- ＭＺ2000

￥3,500 Ⓣ

せまりくる地底人（グルバー）を爆弾をつかってドンドンやっつけるGAME。人間はボンブ・マンを操り、地面を掘り進み、爆弾を落とす。爆弾は何かにぶつかると爆発し、その爆風にグルバーをまきこんでやっつける。爆弾はただ数を打てば良いというもんじゃなく、下手をすると自分も爆風にまきこまれるので注意しよう！



●ＡＤＵＣＫ

- Ｘ１Ｄ
- ３インチＦＤ

￥4,800

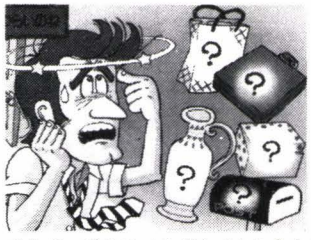
「当社からぬすまれた、極秘マイクロフィルムをうばい返してほしい」という仕事がきみのところにまいこんできた。早速、君は敵地ＴＯＭ社にせんにゅうした！が…Ｘ１、MZ2200￥3,500

●レーダー大作戦

- ＭＺ2000
- ＭＺ2200

￥3,000 Ⓣ

あなたはＮ国と戦争中のＴ国の総指揮官です。レーダーをうまく使用し敵かん隊を全滅させなければいけない。

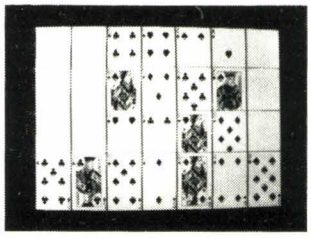


●スパイってつらいのね

- ＭＺ2200
- ＭＺ2000

￥4,000 Ⓣ

ある下っぱのスパイが上司から命令書を受けとった。しかし、飲みすぎて命令書をどこに置いたのか忘れた。二日酔いで頭はガンガンするが命令書をさがしてなんとか任務をすいこうしなければならない。さて……



●プレーラスベガス

- ＭＺ5500　ＰＣ9800Ｆ
- ＦＤ　ＲＡＭ256Ｋ
- 漢字ＲＯＭ

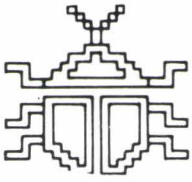
￥6,800

ブラックジャック、バカラ、KENO、ルーレットの４種類のゲームがありルールはすべてラスベガスルールに準拠

ＭＳＸソフト（ＭＳＸラスベガスシリーズ）

神　経　衰　弱	トランプゲームのコンピュータ版です。	￥ 3,000
セブンスタットポーカー	トランプゲームの最高峰コンピュータとのかけひき	￥ 4,000
ル　ー　レ　ッ　ト	ご家庭でラスベガス気分	￥ 3,500
バ　　カ　　ラ	日本のオイチョカブに似たゲームです	￥ 3,500
Ｋ　Ｅ　Ｎ　Ｏ	ビンゴに似たゲームです。	￥ 3,500

尚お客様には、漢字ワードプロセッサー夏目漱石の開発が遅れております事を深くおわび申し上げます。又ソフトテープ、その他商品に万全をつくしておりますが時々ごめいわくをお掛けすることがあります事を重ねてよろしくお願い致します。



◎送料は一律500円かかりますのでお願い申し上げます。◎注文は現金書留で品名及び電話番号記入の上お願い申し上げます。
◎注文の時はコンピューターの機械等くわしく記入の上お願い申し上げます。

株式会社 ウスキパソコンセンター
〒930 富山市総曲輪3の6の3 ☎(0764)21－4181㈹(内線25)
●営業時間：午前10時➡午後５時
●定 休 日：毎週木曜日

【　漢字が誰にでも簡単に使えるシステム　】

```
┌(　ワープ　mini　)┐
│                  │
│  1 ： 文書作成   │
│  2 ： 文書再生   │
│  3 ： 文書記憶   │
│  4 ： 文書印字   │
│  5 ： 書式指定   │
│  6 ： 作表機能   │
│  7 ： 補助機能   │
│                  │
└(ディスク版　メニュー)┘
```

I/Oデータ機器の漢字ROM又は、MZ-1R13を使用して1度に2000文字の[illegible]を作成することが出来ます。ディスク版ではシークレット機能があり1ディスケットに40ファイル(8万文字)記録出来、単語(6052語)、熟語(6000語)ファイルが付いております。印字は、拡大、縮少、横書、縦書の4種類で強調印字を文中で指定することが出来、文字間隔、行間隔が指定出来ます。又印字シュミレーションで印字内容をテレビ画面で見ることが出来ます。その他機能としては、計算機能、センタリング、右づけがありディスク版では外字登録も出来ます。書式指定で使用するプリンタを指定することが出来ます。カナ漢字変換ROM対応MZ-80Bの文書作成画面は、横20文字で、MZ-2000の文書作成画面は、横40文字で行い左右スクロールは一行の印字数になり最高50文字で、上下スクロールは40行～100行です。

上記旧バージョンをお持ちの方は、￥1，500円同封の上当社までディスク又はカセットをお送り下さい。

【　KF1　】　縦横集計プログラム　　(対話型)

項目数: 最大16項目　　項目種: 文字/数値/パス
文字数: 20文字以内　(MZ-80B)　16×16 ドット
　　40文字以内　(MZ-2000)　16×16 ドット
数値数: 倍精度　16桁以内　16×8 ドット (半角文字)
行　数: 2万文字+総項目文字数
項目間計算　: 四則演算、合計、平均、標準偏差、順位、自動番号
行　間計算　: 四則演算、合計、平均、標準偏差、整数化
印　字　　: カード形式
　　表　形式 (ケイ線付・無)
　　タイプライタ形式
　　宛名印刷　: シール印刷、封筒印刷
検　索　　: 2重条件検索方法
　　検索条件 …… 無条件、同じ、大きい、小さい、含む、含まない　(メニュー画面)
指定プリンタ --　MZ-80P6、MZ-1P07　　指定ディスク --　シングル・ダブルディスク

```
┌【　漢字ファイル　】┐
│                    │
│  1 ： 項目　処理   │
│  2 ： ファイル 処理│
│  3 ： データ　処理 │
│  4 ： 表示　処理   │
│  5 ： 印字　処理   │
│  6 ： 計算　処理   │
│  7 ： 並び替え 処理│
│                    │
│ 番号を入力して下さい。?=戻 │
└────────────────────┘
```

【　オールマシン語による超高速漢字住所録　】

検索速度　-　1000名中、1人検索時間最高50秒以内
並び替え　-　1000名並び替え時間　5分前後
処理人数　-　カセット版　150名　ディスク版　1000名
項　目　-　氏名、フリガナ、電話番号、住所1、住所2、住所3
　　郵便番号、備考1、備考2、備考3、備考4
熟　語　-　1200語(県名、市名700語登録済)学習機能付
外　字　-　95文字
宛名印刷　-　シングルシール、ダブルシール、ハガキ印刷
検　索　-　2重条件検索方法
検索条件　-　無条件、同じ、大きい、小さい、含む、含まない
プリンタ　-　MZ-80P6、MZ-1P07
ディスク　-　シングルディスク、ダブルディスク
住所一覧　-　表示、印字は項目を自由に設定することが出来ます。(ディスク版)
構　成　-　MZ-2000(MZ-2200)+グラフィック3ページ+ディスク(カセット)+プリンタ
　　+PIO-3055(MZ-1R13)
　　MZ-80B+グラフィック1ページ+ディスク+プリンタ+PIO-3055(MZ-1R13)

```
┌【　漢字住所録　】┐
│                  │
│  1 ： データ登録 │
│  2 ： データ表示 │
│  3 ： データ印字 │
│  4 ： 宛名印刷   │
│  5 ： 並び替え   │
│  6 ： 補助処理   │
│                  │
│ 番号を入力して下さい。?=戻 │
└──────────────────┘
```

(　ディスク版　メニュー　)

ワープmini	[V5.1T]	MZ-80B	カセット	22,000円
ワープmini	[V5.1D]	MZ-80B	ディスク	49,800円
ワープmini	[V6.1T]	MZ-2000・MZ-2200	カセット	22,000円
ワープmini	[V6.1D]	MZ-2000・MZ-2200	ディスク	49,800円
漢字ファイル KF1	[V5.2D]	MZ-80B	ディスク	68,000円
漢字ファイル KF1	[V6.2D]	MZ-2000・MZ-2200	ディスク	68,000円
漢字住所録(New)	[V5.3D]	MZ-80B	ディスク	33,000円
漢字住所録(New)	[V6.3T]	MZ-2000・MZ-2200	カセット	15,000円
漢字住所録(New)	[V6.3D]	MZ-2000・MZ-2200	ディスク	33,000円

〒416
静岡県富士市長通104-3
スガヤ無線販売会
電話(0545)61-1417

-取扱い店募集中-
お求めは全国マイコンショップ又は当店あてに現金書留(送料サービス)に機種名を書いてお送り下さい。
・営業時間: 午前9時→午後7時　　・定休日: 毎週水曜日

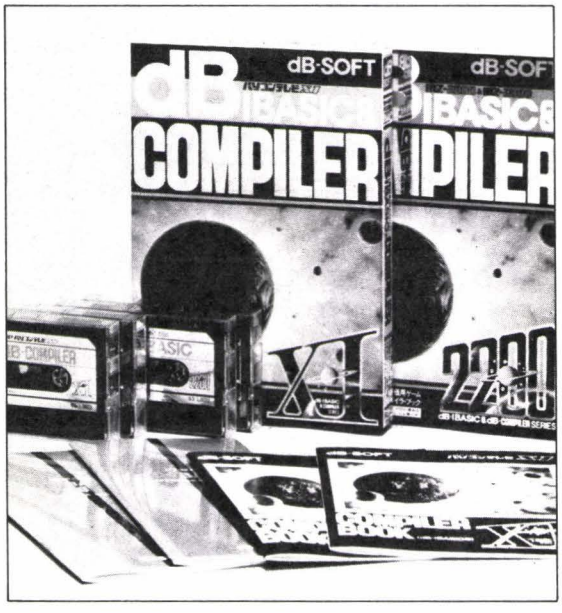
dB-SOFT

話題沸騰!! 驚異のフリーエリア32Kバイト!

dB-BASIC

「速い!」「広い!」デービーファン既に7万人突破!!

dBコンパイラ 本物思考BASICコンパイラ誕生!
君の頭脳は数10倍の力を発揮する



パソコンテレビX1・mz-2200(MZ-2200)用

■dB-IBASIC(整数型ベーシック)

- ●扱える数値が−32768～32767までの整数。
- ●dB-BASICの姉妹言語として、扱えるコマンド数が70%まで同じ、更にdB-BASICとの違うコマンドは付属マニュアルで区別。dB-BASICで開発したプログラムも多少の改良でコンパイル可能。
- ●高密度なゲーム作りに欠かせない広いフリーエリアを41KB確保。
- ●ゲーム向きのコマンド数を更に増加し、ゲーム開発用言語として強化。
- ●プログラムしやすく、デバックも簡単。

■dB-コンパイラ

- ●IBASICで作成したプログラムが、10数倍のスピードのマシン語プログラムに変換。

PC-8801MKII用好評発売中!!

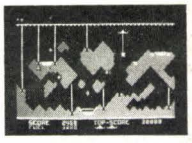
評価用ゲーム

●dB-IBASICで作成したプログラムとコンパイルしたプログラムで、速さの違いを確認できます。

■X1用

「レーザープラネット」

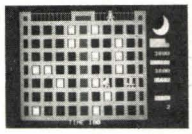
レーザ光線をかいくぐり、うまく基地に到着できるか。



■MZ-2200用

「快盗ストルン」

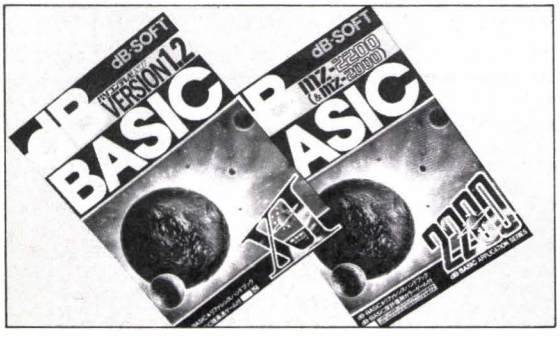
愉快な快盗、見事財宝を盗み、ビルから逃げられるか。



dB-IBASIC/dB-コンパイラ/評価用ゲーム/リファレンスブック/コンパイラブック

各6,000円

dB-BASIC 個性が生きる、高速グラフィック処理!
dB-BASICは君の頭脳の一部となる



パソコンテレビX1・mz-2200(MZ-2200)用

■X1用 Ver.1.2

dB-BASICバージョン1.0に続きX1用バージョン1.2発売中!!

- ●dB-BASICを更にバージョンアップ。LINE機能の強化と、グラフィックのGET・PUTコマンドを追加することによって、グラフィック機能を強化。
- ●XIのPCG機能を利用するとグラフィック効果がグ～ンとアップ。
- ●フリーエリアを32KBまで確保し、また必要に応じて独自のコマンド定義も可能。
- ●高速演算ルーチンの採用により、従来の各種ベーシックよりスピーディな処理が可能。

■MZ-2200(2000)

- ●疑似PCG機能の搭載により、8×8ドット単位でキャラクターを設定。

dB-BASICプログラムカセット/評価用ゲームカセット/リファレンスブック

各5,000円

TOTAL SOFTWARE HOUSE

ぼくの名前がきまったヨ！
「バッくん」です。
かわいがってヨ〜ン！

更に機能を強化して、パソコンテレビXI用好評発売中!!

簡易言語 NEW VIP

最強の簡易言語!!累計30万本突破!!

dB-SOFT「NEW-VIP」(簡易言語)はBASIC知識を一切必要としません。対話型データ入力方式の採用により、初めてパソコンに触れる方でも簡単操作。24のコマンドを押すだけで実行命令を入力でき、スピーディーに処理。また、各種機能は目的・用途に合わせた組み替えが自由…ユーザーズサイドに立って開発された汎用型簡易言語……これが「NEW-VIP」です。

並べ替え自由、ソート機能

データを五十音順や年令順等に並べ替えるソート機能を搭載。数回の条件入力だけで、素速くソーティング、条件が複数項目になっても実行が可能。

■年令によるソート例(ソート前)

```
タカハシ ジュン  サマ  18  M  サッポロシ トヨヒラ.....  ×××
チバ ヒトシ      サマ  20  M  サッポロシ チュウオウク キタ5ニシ4
ハヤシ ミキ      サマ  19  W  サッポロシ シロイシク 1−2★ −−−MS
モリアイ ヤスシ  サマ  22  M  サッポロシ キタク キタ★★ ヒカシ★★
マツイ アヤコ    サマ  19  W  ホッカイドウ オタルシ ?????
```

(ソート後) ⇩

```
モリアイヤスシ   サマ  22  M  サッポロシ キタク キタ★★ ヒカシ★★
チバ ヒトシ      サマ  20  M  サッポロシ チュウオウク キタ5 ニシ4
ハヤシ ミキ      サマ  19  W  サッポロシ シロイシク 1−2★ −−−MS
マツイ アヤコ    サマ  19  W  ホッカイドウ オタルシ ?????
タカハシ ジュン  サマ  18  M  サッポロシ トヨヒラク ..... ×××
```

組み合せ自由、高速検索機能

検索条件(名前・年令・住所等)を設定することで、迅速かつ正確に検索を実行。複数条件による検索も可能、入力方法は簡単。文字を検索条件として実行させることもOK!

```
0000000 データ ノ ケンサク (2) 0000000
NO  コウモクメイ          モンスウ
 1  ナマエ                =15
 2  ユウヒンバンコウ      =6
 3  ジュウショ1           =25
 4  ジュウショ2           =25
 5  テンワバンコウ        =15
 6  ネンレイ              =2
 7  セイベツ              =1
ケンサク ジュウケン ヲ イレテクダサイ
1="ヤマダ"   3="サッポロ"   7="M"
```

⇩

```
ナマエ          ヤマダ ケイイチ
ユウヒンバンコウ  062
ジュウショ1      サッポロシ トヨヒラク ヒラキシ
ジュウショ2      2−14 スターMS 101
テンワバンコウ   011−813−0001
ネンレイ         21
```

あらゆる作業に大活躍!!

顧客、仕入管理、名簿、統計処理、また家計簿から野球の選手管理に至るまで、幅広く活用!!

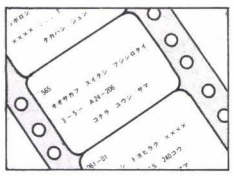
その他の"NEW-VIP"機能

- マシン語記述でスピード処理。豊富なフリーエリア43Kバイト確保
- 対話型データ入力方式
- 大きな分類能力
- 特にXI用には関数計算機能をサポート

(機種により、若干機能が異なります。)

宛名書き

検索を行うことが可能で、内容・用途に合った効果的なDM発行に便利です。



宛名シール POSTACK(ポスタック)

NEW-VIPの印字フォーマットに合わせたdB-SOFTオリジナルタック(宛名)シールです。
(500枚ロール×2巻)…3,600円

■全国通信販売のご案内(送料無料サービス!!)

全国の皆様へ当社のオリジナルソフトウェアを通信販売しております(送料無料)。ご注文は現金書留か銀行振込でお願いします。振込先「たくぎん札幌駅前支店 普通053-053」。銀行振込の場合、住所・氏名・電話番号・商品名・対応機種名・個数をあらかじめハガキでご連絡ください。宛先、当社「Oh!MZ」係まで。(尚、資料をご希望の方は、右下の資料請求券を官製ハガキに貼ってお申し込みください)

NEW-VIPシリーズ 好評発売中

■テープ版 … 各12,800円(本体プログラム・ソートプログラム・サンプルデータ・コマンド下敷・操作マニュアル)
●MZ-2200(2000)用●MZ-80B用●MZ-1200(MZ-80K/C)用●MZ-700用●パソコンテレビXI用

■FD版……各34,800円(本体プログラム・サンプルデータ・コーディングノート・コマンド下敷・操作マニュアル)
●MZ-2200(2000)用●MZ-80B用

発売元 デービーソフト株式会社
〒060 札幌市中央区北3条西2丁目カミヤマビル☎011(251)7462

資料請求券 Oh!MZ 5月号

Enjoy Computer!

周辺機器特売!

あなたのパソコンへ心をこめて

＊15回払いは一例です。
分割お支払いは36回までありますのでご自由にお選び下さい。

ソフトが豊かな人気パソコン。特価周辺機器でグレードアップも楽しめます。

機種	定価	現金特価	初回のお支払い	2回目以降のお支払い(14回)	分割手数料(15回払)	支払い合計金額	
○パソコン							
①NEC PC-9801F2	￥398,000	￥358,000	￥23,280	￥28,000	￥57,280	￥415,280	
②富士通FM-8	￥218,000	￥69,000	￥10,040	￥5,000	￥11,040	￥80,040	家庭用TVアダプターサービス
③NEC PC-8001MKII	￥123,000	￥116,000	￥9,960	￥8,900	￥18,560	￥134,560	
④PC-8801MKII model30	￥275,000	￥239,000	￥25,240	￥18,000	￥38,240	￥277,240	
⑤NEC PC-6601	￥143,000	￥120,000	￥13,200	￥9,000	￥19,200	￥139,200	
⑥シャープMZ-2000	￥218,000	￥98,000	￥15,680	￥7,000	￥15,680	￥113,680	グラフィックRAMボード、RAM2P、3P
⑦富士通FM-7	￥126,000	￥115,000	￥21,400	￥8,000	￥19,200	￥133,400	家庭用TVアダプターサービス
⑧カシオ FP-1100	￥128,000	￥59,800	￥13,360	￥4,000	￥9,568	￥69,368	
F・D・D							
シャープMZ-80BF	￥298,000	￥128,000	￥11,280	￥9,800	￥20,480	￥148,480	I/Oカード、ケーブルサービス(￥46,700)
シャープMZ-1F07	￥158,000	大特価提供中! ￥134,000					他にサービス有
NEC PC-80S31	￥168,000	￥142,800	￥11,648	￥11,000	￥22,848	￥165,648	I/Oカード、信号ケーブルサービス(￥19,000)
プリンター							
シャープMZ-80P4B	￥281,000	￥155,000	￥13,200	￥11,900	￥24,800	￥179,800	I/Oカード、ケーブル付
シャープMZ-80BP5	￥143,500	￥75,000	￥7,168	￥6,100	￥12,768	￥92,568	I/Oカード、ケーブル付
シャープMZ-1P07		￥79,800	￥6,920	￥6,300	￥13,120	￥95,120	I/Oカード、ケーブル付
シャープMZ-1P01	MZ-700用カラープロッタープリンター定価￥39,000を特価￥22,000						

●コンピュータ、各種周辺機器最新製品も全て揃っています、お問合せ下さい。

お支払方法／均等払い、ボーナス併用払い、ボーナス一括払い、3回～36回までの分割払い等ございます。毎月銀行からの口座振替とさせて頂きます。

東芝ファインカラーディスプレイ

●PA-7165(14")定価￥98,000を
大特価￥69,800(接続ケーブル付)
ドット間トリオピッチ0.38mmのノングレアハイコントラストカラーブラウン管!回転角度調整機構付

《パソピア漢字ROM PAC2》

●PA-7247定価￥40,000を特価￥19,800

《パソピアRAM PAC2》

●PA-7240(4K) 定価￥14,000を 特価￥9,800
●PA-7242(16K)定価￥28,000を特価￥19,600
●PA-7244(32K)定価￥40,000を特価￥28,000
●PA-7248(64K)定価￥56,000を特価￥39,200

《パソピア用プリンター》

●PA-7253定価￥139,000特価 ￥98,800
(ケーブル付) 漢字プリントOK!

カシオ漢字ROMパック

●FP-1032K(FP-1000/1100用)
定価￥35,000を 特価￥22,800

カシオフロッピーディスク

●1020FD(デュアルタイプ、FP-1000/1100用)
定価￥298,000を 特価￥138,500

シャープMZ-2200特別セット

MZ-2200+グリンモニター+データレコーダ
(MZ-1T02)……大特価￥119,000

MZ2000/MZ2200用周辺

●漢字ROMボードMZ-1R13
定価￥41,800を特価￥35,500
●漢字ベーシックMZ-021
定価￥5,000を 特価￥4,250
●F-DOS MZ-2004
定価￥50,000を特価￥42,500
●グラフィックボードMZ-1R01
定価￥39,000を特価￥27,300
●グラフィックRAM 2ページ、3ページ
定価￥16,000を特価￥11,200

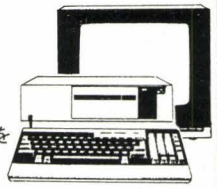
MZ-1U01 MZ-2000用拡張I/Oポート

MZ-2000の利用価値をさらに拡げたい、という方に当店ならではのビックリ値で大放出!
正価￥37,000を
特価￥19,000
(送料サービス)

シャープMZ-2000
大特価 ￥69,800

パソコンテレビX-1

CZ800 C/D
システム価格￥268,000を
超特価 ￥198,000



パソピアー7

本体価格￥119,800
特価￥82,000

富士通FM-8用漢字ROMセット

●MB22002/22003 定価￥40,000 特価￥27,000

通信販売もご利用下さい。全商品、特価でクレジット取扱います。 特価品は送料着払でお願いいたします。

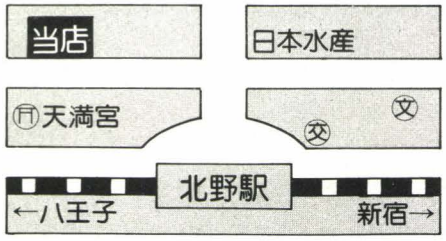
ご注文は在庫確認の上、現金書留又は、銀行振込みでお申込み下さい。全製品保証付、中古品は扱っていません。

創業26年信用をモットーに信頼ある商品を安価にてお届け致します。

(有)アマノ電器サービスセンター

〒192 東京都八王子市北野町560-5
＊京王線北野駅下車徒歩3分
八王子駅よりバス北野駅前下車

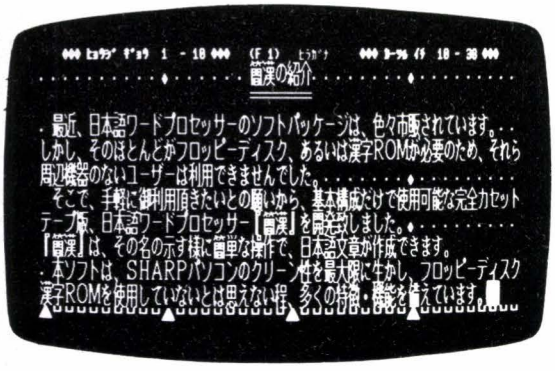
☎0426(45)3001(代)夜間可
《水曜定休日 営業時間/AM10:00～PM7:00》



＊本誌発行時においてよりお求めやすい価格に変更されている場合がございますので価格を再度電話にてご確認下さい。

XI・MZファン待望のワープロ 真打ち登場!!

その名も 簡漢 CAN CAN



（XI用の画面です）

- 漢字ROM不要
- フロッピーディスク不要
- オールマシン語による高速処理
- JIS第一水準の漢字・非漢字OK(注1)
- 新聞紙面の漢字カバー率99%以上(注2)
- 拡張I／Oボックス不要(注1)
- 漢字プリンター不要(注2)

簡漢の主な機能

●ひらがなモード	●自動スクロール	●外字登録	●オールクリア	●センターリング
●カタカナモード	●リピート	●外字呼出し	●枠組み	●倍角文字印字
●英数字モード	●ブロック表示	●短文登録	●枠組み自動補正	●印字文字間隔指定
●グラフィックモード	●レイアウト表示	●短文呼出し	●アンダーライン	●印字行間隔指定
●カナ漢字変換モード	●水平タブ設定	●任意文字数挿入(注3)	●アンダーライン自動補正	●任意行間印字指定
●画面ホーム	●水平タブレーション	●任意文字数削除(注3)	●文書ロード	●用紙サイズ指定(注1)
●改行	●特殊記号・単位表示	●一行クリア	●文書セーブ	●自動紙送り指定(注1)
●カーソル移動	●JISコード入力(注1)	●ブロッククリア	●文書ベリファイ(注2)	●印刷枚数指定(注1)

(注1) Xi・XIcのみ　(注2) MZ-2000・2200のみ　(注3) MZ-2000・2200は一行(40文字)に固定

印字見本

実物大です

CZ-80PKの印字見本

SHARPパソコンテレビX1シリーズ
日本語ワードプロセッサー 簡漢
機能も価格も二重マル。 9，800円

MZ-1P07の印字見本

SHARP MZ-2000・2200
日本語ワードプロセッサー 簡漢
気軽なワープロII 9，800円

機器構成＆価格

パーソナル コンピューター	プリンター	価格
XI 又は XIc (XIの場合はグラフィックRAMが必要です)	漢字プリンター CZ-80PK	9,800円 (わかりやすいマニュアル付)
MZ-2000 又はMZ-2200 (MZ-2000の場合はグラフィックRAM1・2・3が必要です)	ドットプリンター MZ-80P6又は MZ-1P07	

ソフト開発
総発売元
久留米マイコンセンター
〒830 福岡県久留米市通東町3－4
TEL (0942) 39－2404

代理店募集中

●お気軽にお問合せ下さい。

※通信販売は機種名を明記の上、送料300円を加算し現金書留でお申し込み下さい。

ユーザーのための
機械語ユーティリティー

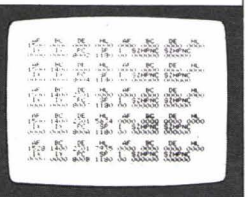
機械語便利ソフト4種

オールマシン語

あなたのマイコンは眠っていませんか？高いお金を出して買ったマイコンです。徹底的に使ってみましょう。機械語はマイコンの基本です。すべてのプログラム（BASIC,COBOL,FORTRAN,……他）は機械語で作成されています。あなたも、機械語を勉強して、マイコンの動作原理を学んでみましょう。基本がマスターできれば、すべてに応用できます。これから紹介する4種は、今から機械語を始めてみようという方、もっとマイコンを徹底的に使いこなしてみたいと思っている方、BASICに飽きた方、など初心者から中級者まで使用できる便利なソフトウエアです。1度、使ってみて下さい。

機械語トレーサー
&デバッガー

好評発売中



機械語動作テストツール

Z80 RELOCATABLE TRACER
リロケータブル　トレーサー

BASICに比べると、機械語プログラムは、それ自身ではエラーチェック機能がないため、バグを発見するのが大変です。そこで、効率的に機械語プログラムの開発ができるように、又初心者はZ80機械語の動作を理解できるように、使いやすさを徹底的に追求したZ80リロケータブルトレーサーです。ブレークポインタ方式のデバッガ機能、1命令ずつシミュレートするトレース機能の2つの機能を持ち,自分自身のリロケート機能を利用して、あらゆるアドレス(モニター部、ROM部は除く)でのトレースを可能にしています。

特　徴

① このプログラムは、リロケート機能が付いているので自由なアドレスに配置でき、任意なアドレスで機械語のトレースが可能です。
② Z80シミュレータを内蔵していますので,1命令ずつトレースすることができます。
③ ブレークポインタは32個まで、ループカウンタはそれぞれのアドレスに対して、最大255回までセットすることができます。
④ スクリーンエディット機能をフルに使っていますので、各レジスタの修正、フラグ(ビット単位)の修正が簡単にできます。
⑤ プリンタ出力機能が付いていますので、トレースした結果などをプリンターに打ち出すことができます。
⑥ 逆アセンブル機能が付いています。
⑦ リロケータブルプログラムが付いているので、各種機械語を自由なアドレスにリロケートすることができます。
⑧ メモリーライト、メモリーダンプ、ブロック転送機能も付いています。

①MZ-700用
②MZ-1200,80K/C用
③MZ-2200,2000,80B用
④パソコンテレビX1用
⑤PC-8001,8001mkⅡ,8801(N-BASICモード)用
⑥PC-8801,8801mkⅡ用

テープ
各￥4,000　〒無料
(マニュアル付)

逆スクロールする
ラベル付
逆アセンブラー

好評発売中

機械語解析ツール

SCROLL DIS-ASSEMBLER
スクロール　ディス　アセンブラー

スクロールディスアセンブラーは、巻物のように上下に自由にスクロールするラベル付Z80逆アセンブラです。

機械語は、普通のダンププログラムでは16進数の表示しかできないので、入力した機械語の確認や、他の機械語プログラムの内容解析が大変です。そこで、このソフトウエアを使用すると、16進数の機械語を分かりやすいニーモニック形式で、又ラベル付きで表示してくれます。また、このプログラムは機械語で作成されているので、高速に解析でき、1つのキー操作で簡単に逆スクロールもできます。

特　徴

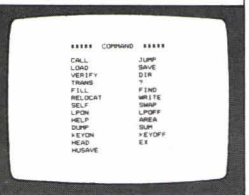
① 上下に自由にスクールするので、機械語の解析が楽にできます。
② スクロールスピードを10段階に調整することができます。
③ ラベル付ですので、リストを見やすく出力することができます。
④ リロケート機能が付いているので、自分自身を自由なアドレスに配置することができます。
⑤ オフセット機能が付いているので、他機種(CPUはZ80)のどこのアドレスのプログラムでも、正常に逆アセンブルすることができます。
⑥ プリンター出力機能、アスキーコード出力機能付。
⑦ データエリアを調べて設定することによって、そのエリアはデータとして逆アセンブルリストを出すことができます。

①MZ-700用
②MZ-1200,80K/C用
③MZ-2200,2000,80B用
④パソコンテレビX1用
⑤PC-8001,8001mkⅡ 8801(N-BASICモード)用
⑥PC-8801,8801mkⅡ用

テープ
各￥4,000　〒無料
(マニュアル付)

多機能な機械語
モニター

好評発売中



機械語入力ツール

SELF RELOCATABLE MONITOR
セルフ　リロケータブル　モニター

あなたは、マイコン雑誌から機械語プログラムを打ち込む時、入力方法が分からなかったり、入力ミスをしたり、モニタープログラムを使いにくく思ったことはありませんか？

セルフリロケータブルモニターは各マイコン雑誌から、機械語プログラムを打ち込むための使いやすさを徹底的に追求した機械語モニタープログラムです。

特　徴

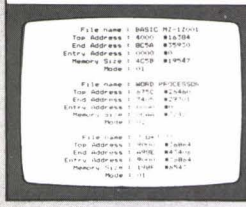
① このプログラムは、リロケート機能が付いているので自由なアドレスに配置でき、任意なアドレスへの機械語の入力、LOAD、SAVEができます。
② 機械語を打ち込む時、1行ずつチェックサムを確認しながら入力ができます。
③ 6~7種類のチェックサム(8バイトごと、16バイトごと、たてよこ(1,2の機種を除く)、128バイトごと、256バイトごと)を出すことができます。
④ キーボードの一部分を16進キー配置に変えて、機械語の入力が素速くできます。
⑤ スクリーンエディタを多くのコマンドで使っているので、修正が簡単にできます。
⑥ プリンター出力機能が付いています。
⑦ 16進、10進、2進の混合加減算機能が付いているので、アドレス計算などができます。
⑧ サーチ機能で機械語の検索ができます。
⑨ リロケータブルプログラムが付いているので、各種機械語を自由なアドレスにリロケートすることができます。
⑩ テープにセーブされている機械語の各アドレスを調べる機能も付いています。
⑪ 約24種のコマンドをもった多機能機械語モニターです。

(除PC-8001用は、機械語をテープにSAVEする時、ファイルネーム、スタートアドスも記録できるようになっていて、SAVEされた、テープはROMモニターのLコマンドでも、PC-8801モニタROMの"R2:RETURN"コマンド(600ボー)でもLOADできます。)

①MZ-700用
②MZ-1200,80K/C用
③MZ-2200,2000,80B用
④パソコンテレビX1用
⑤PC-8001,8001mkⅡ 8801(N-BASICモード)用

テープ
各￥5,000　〒無料
(マニュアル付)

好評発売中



機械語テープ編集ツール

CASSETTE SAVER
カセット　セイバー

機械語をテープにSAVEしている時、ファイルネーム、アドレスなどが分からなくなったことはありませんか？

カセットセイバーは、カセットテープに記録されている機械語のバックアップを簡単に作成したり、カセットテープの情報(ファイルネーム、各アドレスなど)を調べたり、別のテープに編集しなおしたりできるソフトウエアです。

※編集機能について……本体のRAMをフロッピーディスクのように使用して、機械語、BASIC、データなどを空いているRAMエリアに次々にLOADして、それらのファイルネームを変更したり、順位を入れ換えたり、各アドレスを変更したりして、まとめてあるいは部分的にあるいは何回もテープにSAVEすることができます。

特　徴

BASICプログラムは、リストを表示したり、修正したり、LOAD、SAVEが簡単にできますが、機械語はアドレスが関係しているため、分からなくなると、修正したり、SAVEしたりできません。そこで、このソフトウエアを使用すると、簡単に機械語のファイルネーム、トップアドレス、エンドアドレス、エントリーアドレス、ファイルモードなどを調べたり、LOAD、SAVEしたりすることができます。また、ロードアドレスを変更することもできます。

(ファイルネーム、トップアドレス、エントリーアドレス、ファイルモードの変更。ファイル順位の入れ換え。ファイルのさく除。ファイル一覧表の出力。ファイルのLOAD、SAVE。他。プリンター出力機能付。)

①MZ-700用
②MZ-1200,80K/C用
③MZ-2200,2000,80B用
④パソコンテレビX1用

テープ
各￥3,500　〒無料
(マニュアル付)

★当社の便利な機械語ユーティリティソフトウエアは、次の3つの特徴を持っています。★

1. 特殊なモニターは使わず、標準装備のモニター部を利用しているので、使いやすくなっています。
2. 処理したい機械語とアドレスが重なっていてもよいように、リロケート機能を付けて自分自身のアドレスを自由に変更できるようになっています。(カセット・セイバーはアドレスとは無関係ですので、あえて付けていません。)
3. 約4～6KBの機械語で作成されているので、処理が高速で、フリーエリアを多く確保することができます。

(株) マイクロラブ

お求めは、お近くのマイコンショップ、又は通信販売でお願い致します。
通信販売：プログラム名と機種名と電話番号を明記の上、現金書留又は郵便振替でお申し込み下さい。

〒850 長崎市五島町7—17 高島ビル
☎0958-27-3725(代)
郵便振替口座 長崎9—2690

グラフィックスがいよいよ面白くなってきた。

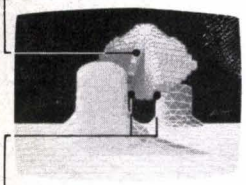
SHARP

豊富なX1のソフトが共通して使えるフルコンパチブル設計

パソコンテレビX1シリーズ

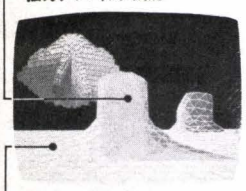
SHARP

好みの図形や記号が自在につくれる
ユーザー定義のキャラクターゼネレータ



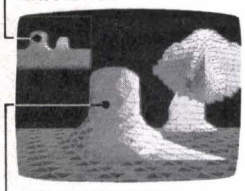
遠近感をもたせたリアルな動きが表現できる
プライオリティ機能

繊細で色彩豊かな表現ができる
強力グラフィック機能



多彩な色づけが簡単にできる
タイリングペイント機能

図形の拡大、縮小そして表示位置を任意に指定できる
座標変換機能



グラフィックの色を瞬時に変更できる
パレット機能

■シャープのエレクトロニクス技術が、パソコンとテレビの独創的なシステムを実現。パソコンの機能がテレビの機能を拡大し、テレビの能力がパソコンの能力を一気に高めました。いまやパソコンはグラフィックアビリティで選ばれる時代。映像表現に新領域をひらいたスーパーインポーズ機能をはじめとしたX1シリーズの新鮮なグラフィックスは、まさにプロのテクニックに迫る充実ぶり。ビギナーからエキスパートまで多くの人々を魅了します。■もちろん、パソコンとしての高機能も見逃がせない特長です。3CPU構成、シャープ独自のクリーン思想を受け継いだメインメモリ64Kバイト、オールRAM。標準入出力装置として、X1、X1Cには自動頭出し機能を備えた2700ボーの高速電磁メカカセットレコーダ、またX1Dには両面倍密度3インチコンパクトフロッピーディスクドライブを装備。そして、こうした信頼のハードウェアに密接に結びついてその機能を最大限に発揮させる強力なソフトウェア。X1シリーズは、究極の8ビットマシンとして、ホームユースからビジネスユースまで幅広い分野への応用が可能な、まさに21世紀のメディアです。

主な拡張用オプション		
フロッピーディスク	CZ-800F	標準価格 ~~198,000円~~
フロッピーディスクインターフェイス	CZ-8FA	標準価格 ~~24,000円~~
ドットプリンタ	CZ-800P	標準価格 ~~142,000円~~
デジタルテロッパー	CZ-8DT	標準価格 ~~89,800円~~
グラフィックRAM	CZ-8GR	標準価格 ~~32,000円~~
漢字ROM	CZ-8KR	標準価格 ~~38,000円~~
拡張I/Oポート	CZ-8EP	標準価格 ~~11,800円~~
RFビデオコンバータ	CZ-8VC	標準価格 ~~15,800円~~

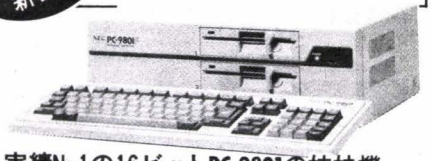
あのX1が選べる3タイプ

有名メーカー 店頭発表価格！ ホビーからビジネス

ソフト2000本

パーソナルコンピュータ
CZ-801C 標準価格 ~~119,800円~~

新製品 NEC PC-9801F

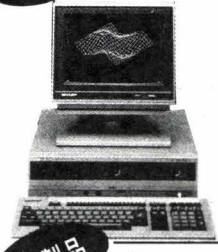


実績No.1の16ビットPC-9801の姉妹機
PC-9801F1(1DD) ~~¥328,000~~
PC-9801F2(2DD) ~~¥398,000~~

新製品 NEC PC-100シリーズ

PC-100モデル10(1DDモノクロ) ~~¥398,000~~
PC-100モデル20(2DDモノクロ) ~~¥448,000~~
PC-100モデル30(2DDカラー) ~~¥558,000~~

新製品 シャープ MZ-5500



MZ-5521 ~~¥388,000~~
●256Kバイト●2DD

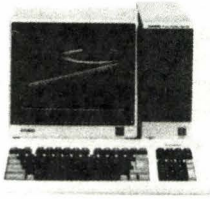
MZ-5511 ~~¥288,000~~
●256Kバイト●1DD

MZ-5501 ~~¥218,000~~
●128Kバイト

新製品 パソコンテレビ X-1

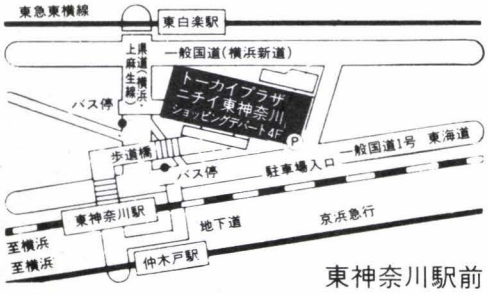
パソコンテレビX1 ~~¥155,000~~
パソコンテレビX1C ~~¥119,800~~
パソコンテレビX1D ~~¥198,000~~

新製品 カシオ FP-3000



TYPE-1 ~~¥148,000~~
TYPE-2 ~~¥198,000~~
(いずれも本体価格)

- 快速16ビット
- 快速ペイント120色
- 快速スーパーインポーズ
- 快速マルチスクリーン48画面
- 快速簡易言語パビルス-II
- 快速カナ漢字変換



東神奈川駅前

■お支払いは便利なクレジットで
ボーナス一括払い実施中！

●お支払い例 (¥215,000お買上げの場合)
国内信販でご利用の場合
月々わずか ¥23,650×10回
月々わずか ¥12,680×20回

●住友カード●ダイヤモンド●ユニオン
●日本信販カード●ミリオンカード●日本ダイナーズ●JCB……取扱店

ペロ**トーカイプラザ** ニチイと70の専門店・4F

横浜市神奈川区富家町1番地(東神奈川駅前) TEL.045(421)3005

MZを使うほど、ほしくなる…コスモス岡山の
UTILITY SOFT

CRAZY FORMATTER

MZ-80B
MZ-2000用 **¥8,000**（〒200）

MZ-80B、2000のDISK-BASICで使用するスレーブディスケットのユーザエリアを、従来の"Utility"プログラムで作成したスレーブディスケットよりも60～83.7Kバイトユーザーエリアの増したスレーブディスケットを作成します。

	"Utility"コマンド		CRAZY FORMATTER
	MZ-80B	MZ-2000、2200	MZ-80B、2000、2200
フリーセクター	977	1072	1312セクター
ユーザーエリア	244.25Kバイト	268バイト	328Kバイト

注）シャープ社より発表のフロッピーディスク装置を使用して、

また、付属のコピーコマンドは途中にキズがあったり、フォーマットされていないトラック、セクターが存在して読み込み不可能なトラックセクターがある場合、自動的にそのセクターを読み飛ばしてバックアップコピーを続行します。
（読めなかったトラックセクターはモニターディスプレイに表示されます。）

SUPER MACHINE LANGUAGE

●雑誌のプログラムリストの打ち込みに ●マシン語プログラム作成時の虫取り用に ●マシン語の勉強に…という方におすすめ

MZ-80B、2000用
MZ-80K/C、1200用 **各¥3,800**（〒200）

Super Machine Languageは、シャープ社のマシンランゲージSP-2001を更に使いやすく、機能的なマシン語モニターとして作られました。スクリーンエディターの強化に加え、メモリーダンプのスピードアップ・命令単位でのメモリーダンプ・指定アドレス間のチェックサムの計算・プログラムリロケート・オペコードサーチ・アスキーコードサーチ・メモリーペイント・テープコピー・コールコマンド…等 便利なコマンドも豊富に追加!!

また、MZ-80B、2000用には逆アセンブル機能が、80K/C、1200用にはオープニングメッセージ画面作成コマンドが追加されております。

また雑誌などのプログラムダンプリストの打ち込みもスピーディーに、かつ間違いなく入力できる様、入力時にチェックサムが表示され入力ミスの発見も大変楽になります。

全23コマンドの豊富な機能を持ったマシン語モニターをこの価格で!!

COFFEE BREAK

ディスクユーティリティーシリーズ第3弾!! Disk Backup Utility

MZ-80B、80B2用
MZ-2000、2200用 **各¥9,800**（〒200）

操作は簡単…最強のバックアップ機能!!

ドライブ1にソースディスケットを入れ、ドライブ2に新しいディスケット（イニシャライズしてないディスケットでOK）を入れて、スタートトラック、エンドトラック…などを設定するだけ!!

コピー実行中ディスプレイ上に各トラックに施されているフォーマット状況やセクターの並びなどを表示しながらコピーを進めて行きますので、ディスケットのフォーマットやセクターインタリーブを調べるのにも大変便利!!

その他機能として…

- パラメーターの指定により各トラックのフォーマットを調べる緻密度を指定できます。（たとえば12トラックに施されているフォーマットだけ特別緻密に調べる事も可能）
- トラック、セクター番号が有るのに実際には使用されていないダミーセクターや特殊フォーマットモードを持ったセクターは特定の記号が付けられ表示され一目瞭然。
- フォーマット状況のプリンター出力も可!!

御注意

※フロッピーディスクドライブはデュアルタイプ（2ドライブ）が必要です。

おなじみ、ロングセラーソフト

S-DOM

MZ-80B、B2用、MZ-2000、2200用
MZ-80K/C、1200用 **各¥9,800**（〒200）

S-DOMは日常のプログラム、データーのバックアップからディスケットの内容解析やうっかり"DELETE"してしまったファイルの復活（B、2000用のみ）… 等に幅広し利用できる様に、20数種類のコマンドが用意されたディスクユーティリティープログラマです。

〈一例〉

XFERコマンド：指定したデバイス間でプログラムやデーターをファイル単位で転送します。ディスケット間ではBTX、OBJ、BSD、BRDの転送ができます。カセットテープディスケット間ではBTX、OBJ、BSDの転送ができます。

DUMPコマンド：ディスケットの任意のトラック、セクターの内容を1セクター単位で表示します。（スクリーンエディットでセクター内容の書き替え可）

DATAコマンド：指定した"filename"の属性、スタートアドレス、プログラムサイズ… 等8項目のデーターを表示します。（スタートアドレス、エンドアドレスのわからない機械語プログラムの正体も 一目瞭然!!）

MAPコマンド：ディスケットの使用状況を1セクター単位で表示します。

COPYコマンド：ディスケットのコピーを作成します。（70トラックフルコピー機能を持っておりますので、DISK-BASIC等はマスターディスケットのままコピーできます。）

CNTLコマンド：プリンターにコメントを打ち出したり縮少文字紙送り幅の設定をします。

SAVEコマンド：指定したメモリー内容を、BTX、OBJ、BSD、BRDの好きなファイルモードで、カセット、ディスケットにセーブします。

Mコマンド：任意のメモリー内容を表示します。

その他、LOADS、MASTER、FULL……

シングルディスク対応、実行結果のプリンター打出し、メモリー内容、ディスケット内容の表示は全てスクリーンエディト可能。

WORLD WIDE COMPUTER SUPER SHOP
COSMOS™ 岡山

〒700 岡山市南方5-6-5 今田ビル2F（県営グランド入口向）
TEL（0862）54-7474 ミニファクス（0862）54-7481
年中無休 AM10:00～PM7:00
通販は現金書留又は郵便振替 岡山4-12524 コスモス岡山

High-Powerd王将

新発売

強力 高速 王将 HP

「王将」の思考ルーチンに、皆さまの暖かい御支援のハガキ（約500余通）をもとに、大巾な手直しが加えられ、全く新しい「HP王将」が生まれました。

■画期的エディットコマンドの追加!!
- 詰将棋の場合を設定し、そこから対局可能!

■駒落ち対局可能

■対局を再現可能
データのロードセーブは、HP王将らしく対局のすべてを記録・再生ができるようになりました。更に、左記の項目も可能になりました。
- 今までの対局の再現可能
- 再現途中場面からの差し直し可能
- 修正しての差し直し可能

詳細マニュアル
「棋太平」付

▲カラー写真でなくて残念です。

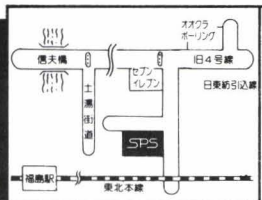
■王将の2倍以上のスピード!
■強くなりました。
■画面にきれいな色がつきました。
（2200/2000はカラーモニター使用時、もちろんグリーンモニターでもOK）

金四千弐百円也

2200/2000用 X-1用

■ポーラースターIIも好評発売中!!
■SPSではひきつづきプログラマーを募しております。王将ぐらいなら作れると思っている人は、どんどん御応募下さい。

お求めはお近くの有名マイコンショップで。
通販希望の方は使用機種名明記の上現金書留でお願いします。
（送料サービス）



シャープコンピューター販売 プログラム作成
マイコンハウス SPS
〒960 福島市太平寺字町の内5-3 ☎(0245)45-5777
FAX(0245)45-1804(GII,GIII)

パートナーショップ
キャリーラボ

信用と実績を誇る

BASIC HOUSE

宇都宮の本格的なマイコンショップ

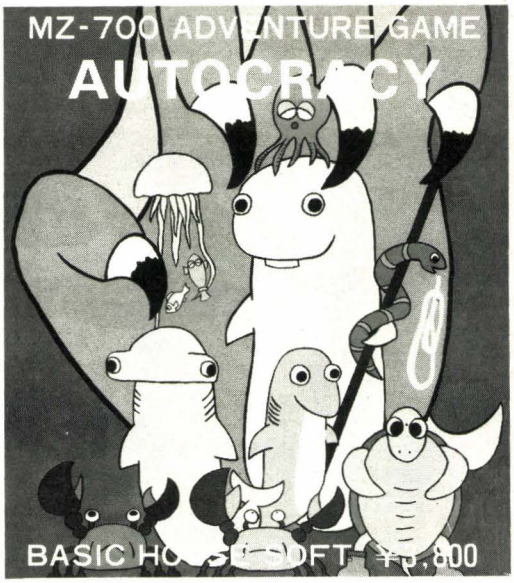
MZ-700はじまって以来の超大作 ADVENTURE GAME

アップル風本格的アドベンチャーゲーム

画面数 120 枚の超大作

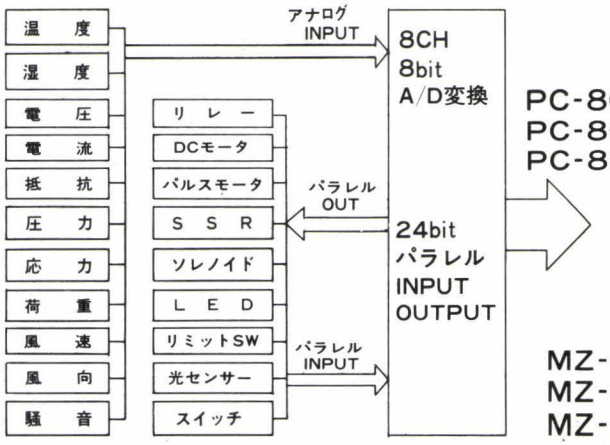
AUTOCRACY

ある朝、あなたは起きてすぐ昨日見た夢を思い出しました。それは不思議な夢で、地球のどこかで独裁政治が行われていて、そこに住む住民たちが、たいへん困っている…という夢でした。あなたは頭を冷やすために散歩しようと思い、家のドアを開け一歩外へ踏み出すとそこは見たこともない所で、まだ夜でした。…うしろをふり返えると、もう自分の家はあとかたもなく消えていました。……田中の角サンもビックリ!!



パソコンで計測制御をしよう

■汎用AIO PIOボードキッド新発売!!　定価15,000(完成品定価19,800)



PC-8001mkII
PC-8801,mkII 用

型番 **KGB-PC1**

MZ-80B, MZ-2000
MZ-2200

型番 **KGB-MZ1**

PC, MZともテスト用プログラム，回路図，説明書付

キット価格￥15,000
完成品価格￥19,800

MZ-2000、80B 設計、デザイナー、アニメーション、ラボラトリーオートメーション等を考えている人!!

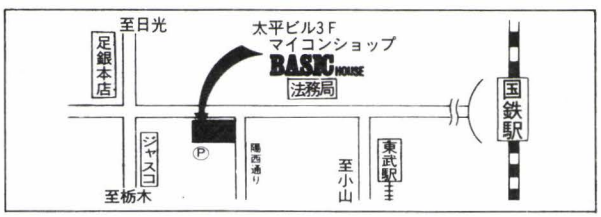
スーパーCAD(FD)

新発売

￥28,000

(ハード)
MZ-2000 CPU
K-510 デジタイザ
MP-1000 X-Yプロッタ

■デジタイザからの入力(直線自由曲線)を使って図面を作成しCRTに表示しX－Yプロッターに作図するデータをフロッピーに登録、呼び出しが出来る。



◆当社の製品はお近くのマイコンショップ等でお求め下さい。また、製品等に関するお問い合わせは直接当社へお願いいたします。

マイコンショップ BASIC HOUSE

〒320 宇都宮市桜3丁目2－17 太平ビル

☎0286－33－1994　3F(株)計測技研

システムソフトウェアコンバータ

BASIC HOUSE

驚異の大ヒット パソコンテレビXI がソフトだけで 続々シリーズ発売中 MZ-2000 MZ-2200 に早変り!!

システムソフトウェアコンバータ 第1弾!!
MZ-2000 MZ-2200 BASIC B6-2213

名機MZ-2000、MZ-2200用のBASICも、パソコンテレビXIで御使用になれますので、MZの豊富なソフトが手直しする事なくそのまま動作します。一部マシン語もOKです。
注意：本ソフトは著作権上MZ-IZ001(￥2,150) MZ-IZ002(￥7,000)が必要です。

機種：X1、X1C　　定価￥3,800

システムソフトウェアコンバータ 第2弾!!
LOGO and PASCAL B6-2217

新発売

パソコンテレビXIで人工知能言語、タートルグラフィックスで思いどおりの図形が描けるMZ-LOGO が動きます。
またB面には話題の言語PASCAL のコンバートソフトが入っています。※本ソフトは著作権上MZ-LOGO(￥9,800)、MZ-IZ004(￥12,000)が必要です。

機種：X1、X1C　　定価￥4,200

システムソフトウェアコンバータ 第3弾!!
新発売
システムプログラムandマシンランゲージ B6-2218

MZ-IZ005(システムプログラム)がA面、MZ-IZ006(マシンランゲージ)がB面にコンバートできる様にシステムソフトウェアコンバータが入っています。マシン語開発ツールですのでアセンブリ言語でプログラミングしたい方には最高です。※本ソフトは著作権上MZ-IZ005 (￥25,000)、MZ-IZ006(￥7,000)が必要です。

機種：X1、X1C　　定価￥4,200

システムソフトウェアコンバータ 第4弾!!
開発中
DISK BASICandCOLOR DISKBASIC B6-2214

MZ-2000・2200用のDISK BASICをパソコンテレビXI で動かすソフトでB面にはCOLOR DISK BASIC(MZ-2Z002)のコンバートソフトが入っています。
※本ソフトは著作権上MZ-2Z001(￥10,000)MZ-2Z002(￥12,000)が必要です。

機種：X1、X1C　　定価￥4,200

システムソフトウェアコンバータ 第5弾!!
倍精度BASIC and 倍精度DISK BASIC B6-2215

新発売

実務ソフトのアプリケーションを御使用になる場合たいへん便利です。※本ソフトは著作権上MZ-IZ003(￥7,000)あるいはMZ-2Z003(￥12,000)が必要です。

機種：X1、X1C　　定価￥4,200

システムソフトウェアコンバータ特別企画!!
N-BASIC B6-2220

開発中

パソコンテレビXIが名機PC-8001に早変り。PC-8001のアプリケーションソフトがXIにて手直しする事なく動きます。
※本ソフトは著作権上XIとPC-8001を持っている方に限ります。

機種：X1、X1C　　定価￥4,800

各種BASICテキストコンバータ　絶賛発売中！

X1	PC-8001	→CZ-800	B6-1483	￥3,800	MZ-700	PC-8001→MZ-700	B5-1483	￥3,800
	PC-8801	→CZ-800	B6-1493	￥3,800		PC-8801→MZ-700	B5-1493	￥3,800
	MZ-80B・2000	→CZ-800	B6-1413	￥3,800		PC-6001→MZ-700	B5-1473	￥3,800
	MZ・80K／C・1200	→CZ-800	B6-1433	￥3,800		X1用逆アセンブラ	B6-2109	￥4,200
	PC-6001	→CZ-800	B6-1473	￥3,800				

開発室速報!!　おもしろまじめな製品を続々開発中！

■パソコンテレビX1で世界のコンピュータ(apple)が動いたらスゴイでしょうね!!
■MZシリーズ、X1シリーズ用5インチDISK(1.0MB)開発中しかも従来のソフトも動くかも!!
■パソコンテレビX1でMZシリーズのI/Oボード(RS-232C、GP-IB、A/Dその他)がそのまま使用できる拡張I/Oスロット開発中(X1用各2スロットづつ)
■ゲームソフトのにがてなBASIC HOUSEが中曾根チャンもビックリする様な超大作を開発中(X1にて)

マイコンショップ BASIC HOUSE TEL.0286-33-1994
(株) 計 測 技 研

電話で深めようマイコンの世界

106 106 106 106 106 無料 106 106 106 106 106 無料 106 106 106 106 106

アコリット マイコン最新カタログ

シャープ パソコンテレビX1

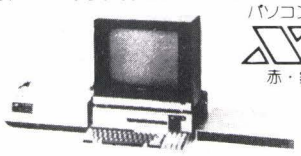
CZ-800C/D ¥ 268,000
48回分割 ¥ 6,550／60回分割 ¥ 5,581

CZ-801C/D ¥ 219,400
48回分割 ¥ 5,362／60回分割 ¥ 4,569

CZ-802C/D ¥ 326,000
48回分割 ¥ 7,967／60回分割 ¥ 6,789

型名	品名	定価/円	48分割	60分割
CZ-800C	本体	155,000→	3,788	3,228
CZ-800D	カラーディスプレイ	113,000→	2,761	2,353
CZ-801C	本体	119,600→	2,922	2,494
CZ-801D	カラーディスプレイ	99,800→	2,438	2,078
CZ-802C	本体	198,000→	4,839	4,123
CZ-802D	カラーディスプレイ	128,000→	3,128	2,666
CZ-8GR	グラフィックRAM	32,000→	800	600
CZ-8EP	拡張I/Oポート	11,800→	300	200
CZ-8KP	漢字ROM	38,000→	900	700
CZ-8FA	フロッピI/F	24,000→	600	500
CZ-800F	ミニフロッピディスク	198,000→	4,700	4,000
CZ-800P	ドットプリンタ	142,800→	3,400	2,900
CZ-8VC	RFビデオコンバータ	15,800→	400	300
CZ-8DT	デジタルテロッパー	89,800→	2,100	1,800

パソコンテレビ X1 赤・銀・白



シャープ MZ-2200シリーズ

MZ-2200 ¥ 128,000
48回分割 ¥ 3,000／60回分割 ¥ 2,600

型名	品名	定価/円	48分割	60分割
MZ-1T02	データレコーダ	19,800→	500	400
MZ-1F07	ミニフロッピディスク	158,000→	3,800	3,200
MZ-1P07	80桁ドットプリンタ	79,800→	1,900	1,600
MZ-1M01	16ビットボードキット	78,000→	1,800	1,500
MZ-1R08	漢字ROMボード	29,000→	700	500
MZ-1D06	カラーディスプレイ	102,000→	2,400	2,000
MZ-1D09	14型TVモニタ	110,000→	2,600	2,200
MZ-80P4B	136桁ドットプリンタ	281,000→	6,700	5,700
MZ-1D15	カラーディスプレイ	72,000→	1,700	1,400
MZ-2Z001	ディスクBASIC	10,000→	200	200

シャープ MZ-3500シリーズ

MZ-3541 ¥ 410,000
48回分割 ¥ 9,800／60回分割 ¥ 8,300

MZ-3531 ¥ 320,000
48回分割 ¥ 7,600／60回分割 6,500

型名	品名	定価/円	48分割	60分割
MZ-1P8	漢字プリンタ	540,000→	12,900	10,900
CE-330P	136桁ドットプリンタ	159,000→	3,800	3,200
CE-333P	136桁ドットプリンタ	310,000→	7,400	6,300
MZ-6P01	I/Oカード	12,000→	300	200
MZ-1F02	ミニフロッピディスク	198,000→	4,700	4,000
MZ-1F03	増設フロッピディスク	90,000→	2,100	1,800
CE-331M	ミニフロッピディスク	350,000→	8,300	7,100
MZ-1F05	フロッピディスク	398,000→	9,500	8,100
MZ-1D03	カラーディスプレイ	163,000→	3,900	3,300

シャープ PC-5000シリーズ

PC-5000 ¥350,000
48回分割¥8,553 ／ 60回分割¥7,288

型名	品名	定価/円	48分割	60分割
CE-510P	熱転写プリンタ	80,000→	1,955	1,666
CE-510F	ミニフロッピーディスクドライブ	198,000→	4,838	4,123
CE-511F	コンパクトフロッピーディスクドライブ	128,000→	3,128	2,665

シャープMZ5500シリーズ

型名	品名	定価/円	48分割	60分割
MZ-5521	本体	388,000→	9,481	8,080
5511	本体	288,000→	7,038	5,998
5501	本体	218,000→	5,327	4,540
MZ-ID10	12型モノクロディスプレイ	418,000→	10,215	8,705
MZ-ID11	14型カラーディスプレイ	113,000→	2,761	2,353
MZ-IR16	増設RAM	30,000→	733	625
MZ-IR11	増設RAMボード	80,000→	1,955	1,666
MZ-IR09	増設ビデオRAMボード	35,000→	855	729
MZ-IF02	ミニフロッピーディスクドライブ	158,000→	3,861	3,290
MZ-IP02	80桁ドットプリンタ	138,000→	3,372	2,874
MZ-IP04	カラーインジェットプリンタ	228,000→	5,572	4,748
MZ-IP06	80桁漢字プリンタ	234,000→	5,718	4,873

シャープ CRTシリーズ

型名	品名	定価/円	48分割	60分割
CZ-140DS	RGB14型カラー	89,800→	2,000	1,700
14M-131C	RGB14型カラー	69,800→	1,600	1,400
14M-142C	RGB14型カラー	99,800→	2,300	2,000
14M-132C	RGB14型カラー	118,000→	2,700	2,300
14M-111C	RGB14型カラー	67,800→	1,600	1,300
14M-141C	RGB14型カラー	69,800→	1,605	1,368
15M-412C	RGB15型カラー	118,000→	2,714	2,313
12M-314C	RGB12型カラー	128,000→	2,944	2,509
CZ-210D	RGB21型カラー	179,000→	4,117	3,508
20M-202C	RGB20型カラー	175,000→	4,000	3,400

MSX 特別セール中

★全てジョイスティックとカセットテープ5本付！
★そのほか各社別のプレゼント付！

日本楽器製造		定価/円 15分割
YIS-803		49,800→3,347
YIS-503		64,800→4,355
SFG-01	FMサウンドシンセサイザユニット	19,800→1,330
YK-01	ミュージックキーボード	17,800→1,196
ゼネラルPCT-50(キーボード付)		146,500→9,298
富士通	FM-X	49,800→3,347
東芝"パソピアQ"	HX-10D	65,800→4,422
	HX-10S	55,800→3,750
日立	MB-H1	62,800→4,220
	MB-H1E	54,800→3,683
三菱	ML-8000	59,800→4,019
ソニー	"HIT-BIT"HB-55	54,800→3,683
松下電器	CF-2000	54,800→3,683
三洋電機	MPC-10	74,800→5,027
	PHC-30	64,800→4,355
キャノン	V-10	54,800→3,683
ビクター	HC-6	64,800→4,355
パイオニア	PX-7	89,800→6,035

※その他周辺機器も全て取扱っております。

シャープ ポケットコンピュータ

PC-1245 ¥ 17,800
48回分割 ¥ 400／60回分割 ¥ 300

PC-1500 ¥ 59,800
48回分割 ¥ 1,400／60回分割 ¥ 1,200

おもしろコンピュータ

カシオ 楽がき PV-2000
¥29,800→¥23,800

セガ コンピュータビデオゲーム
SG-1000 ¥15,000→¥12,000
SC-3000 ¥29,800→¥23,800

任天堂 ファミリーコンピュータ
¥14,800→¥11,800

●コレクトコールでのお問い合せは御遠慮ください。
●この商品のみは送料一500円をたして現金書留にてお願い致します。

お買得コーナー

■カシオ	EP-1100＋ソフト ＝¥65,000
■NEC	PC-8001(下取品) ＝¥50,000
■ナショナル	●JR-100＋RFコンバーター ＋ソフト3巻＝¥30,000 ●JR-200＋プログラム集(1冊) ＋ソフト2巻＝¥40,000
■シャープ	MZ-731→¥83,000

Oh-MZ愛読者特別セール

テレビ		標準価格	48回均等	60回均等
ナショナル	TH21-H55GR	¥219,000	¥5,037	¥4,292
〃	TH19-L55GR	¥188,000	¥4,324	¥3,684
〃	TH15-M55GR	¥139,000	¥3,197	¥2,724
〃	TH14-NV1	¥208,000	¥4,605	¥3,924
ソニー	KV-20XR2	¥199,800	¥4,481	¥3,818
〃	KV-18XR2	¥179,800	¥4,032	¥3,436
ビクター	AV-260	¥290,000	¥6,527	¥5,562
〃	AV-200	¥210,000	¥4,709	¥4,013
東芝	21K691	¥215,000	¥4,821	¥4,108
〃	15K31	¥129,800	¥2,910	¥2,480
シャープ	C-I 19型	¥145,000	¥3,252	¥2,771
〃	C-I 14型	¥ 93,000	¥2,086	¥1,777
ビデオ		**標準価格**	**48回均等**	**60回均等**
ナショナル	NV-630	¥169,800	¥3,613	¥3,203
〃	NV-330	¥159,800	¥3,170	¥3,015
〃	NV-370	¥132,800	¥2,864	¥2,440
ナショナル	NV-800	¥289,800	¥6,665	¥5,680
〃	NV-850HD	¥239,800	¥5,238	¥4,700
ソニー	SL-F5	¥169,800	¥3,808	¥3,245
〃	SL-F3	¥145,000	¥3,169	¥2,771
〃	SL-F17	¥132,000	¥3,036	¥2,587
〃	SL-HF77	¥299,000	¥6,877	¥5,860
〃	SL-HF66	¥249,800	¥5,745	¥4,896
ビクター	HR-D225	¥195,000	¥4,373	¥3,726
〃	HR-D120	¥148,000	¥3,310	¥2,842
日立	VT-16	¥173,000	¥3,880	¥3,306

周辺機器も全て取扱っております。

106 コレクトコールでどうぞ！

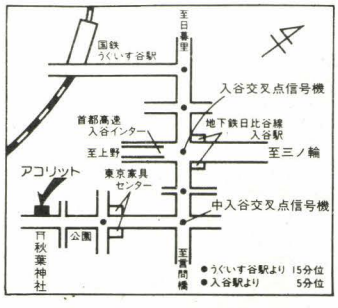
全国どこからでも電話1本(無料)で、商品先取り！お支払いは楽々クレジットで胸キュン！です。
豊富なバリエーションの中から自由に組み合わせてください——

Acolyte (アコリット)
〒110 東京都台東区北上野2丁目12番12号
営業時間／午前11時～午後7時 年中無休
03(844)3361

●他にオーディオ、ビデオ、パーソナル無線も取り扱っています。クレジットは1～60回までOKです。
●初回金のみ、端数金額処理のため、上記金額よりも少し多くなります。
●契約後の取消は規定の手数料を申し受けます。

●この他の新製品の御予約も受付けます。

お支払いは楽な長期クレジット
48回/60回
●現金の場合はお問い合わせ下さい。クレジットは3,000円以上の組み合わせに願います。



わからない、使えないetc……。

パソコン病の方、診察いたします。

パソコンを買ってホコリまみれになっているアナタ。マニュアルを10ページほど読んで何もしていないアナタ。
頭ばかりのコンピュータ知識でとまっているアナタ。同じゲームばかりしているアナタ。
最初はパソコンに夢中だったはず……。中途で放りだしてもったいないですネ!!
一度J&Pのクリニックへいらっしゃいませんか?
パソコンがおもしろくて楽しいコンピュータライフとなるよう、あなたのペースに合わせて
自信をつけてさしあげます。

〈全国全店あなたの情報仲間〉

誰にも使える機種からハイグレード機種まで

ハードウェアコーナー

常に国内、海外の人気パソコンや周辺関連機器をビッグに品揃えて展示しています。
MSX仕様からIBM-5550まであなたのニーズに合ったパソコンがご自由にお選びいただけます。
機種選びのご相談にもお応えいたします。

優秀ソフトが機種別に5,000種の

ソフトウェアコーナー

国内、海外のホビーからビジネスまで優秀な最新のソフトがパソコンの機種別に5,000点を豊富に常備しています。
全国どこからでもお選びいただけます。

あなたが納得できるまでの

各セミナー常設

初めての方に誰れでも分かりやすく楽しいカリキュラムで各パソコン講座を開設しています。
いつでも納得のいくまであなたのペースでご利用できます。

渋谷店	教室 (03) 496-4148
八王子店	教室 (0426)26-4141
町田店	教室 (0427)23-1313
テクノランド	教室 (06) 644-5566
京都寺町店	教室 (075)341-3573
くずは店	教室 (0720)56-8181
四日市店	教室 (0593)54-3366
津店	教室 (0592)26-0111

ビジネスのコンピュータ化に支援する体制。

システムサポート

全国どこからでもご利用ください。

コンサルティング	OA化のご提案をします。(企業内教育・指導、OAシステム提案)
オーダーソフト	ご希望のソフトを作ります。(ビジネス用ソフト開発、特殊用途ソフト開発)
データーサービス	データ入力のお手伝いをします。(データのインプット、ワープロサービス)
アフターサポート	オペレータの教育を引き受けます。(社内インストラクター教育、オペレータ教育)

万全のサポート体制で全国をネットするパソコンの大型専門店J&Pチェーン。

東京 渋谷店	東京 町田店	東京 八王子店	大阪 日本橋 テクノランド
東京都渋谷区道玄坂2丁目28番4号(〒150)	東京都町田市森野1丁目39-16(〒194)	東京都八王子市旭町1番1号八王子そごう7F(〒192)	大阪市浪速区日本橋5丁目6番7号(〒556)
☎(03)496-4141	☎(0427)23-1313	☎(0426)26-4141	☎(06)644-1413
大阪 日本橋 メディアランド	大阪 くずは店	大阪 千里中央店	京都 京都寺町店
大阪市浪速区日本橋5丁目9番11号(〒556)	枚方市楠葉並木町2丁目2－2(〒573)	豊中市新千里東町1-3-204千里サンクワン3F(〒565)	京都市下京区寺町通仏光寺下ル高美須之町549(〒600)
☎(06)644-1613	☎(0720)56-8181	☎(06)834-4141	☎(075)341-3571
名古屋 栄ノバ店	三重 四日市店	三重 津店	新潟 新潟店
名古屋市中区栄3丁目4-5 SAKAE NOVA6F(〒460)	四日市市鵜の森2－1－19(〒510)	津市丸之内31－20(〒514)	新潟市東大通1-3-10三井生命ビル1F(〒950)
☎(052)261-9201	☎(0593)54-3366	☎(0592)26-0111	☎(0252)43-6671

Personal Computer Store

J&P

SHARP

人の思考に近づいた、
「使いこなせる」16ビット。

たとえば、ワープロ。文書をつくりながら画面のスミにレイアウトを重ねて出したり、類義語画面をパッと呼び出したり……。さらに文書をつくる一方で、データの集計・グラフ化をすませ、それをその文書に引用する…というような芸当も、画面を切り換えずに処理できるMZ-5500。高速マルチウインドウやマウス*など、先進の機能がより人間の思考に近いソフト展開を可能にした、まさにビジネスマン待望の一台です。*マウスはオプションです。

データを、もっと深く読みたい。だから複数データを一度に比較できるマルチウインドウ。

経営の分析にしても、書類の照合にしても、人はデータを比較検討しながら新しい発見を得ています。シャープは最大4画面を同時に表示できるマルチウインドウで応えました。うわべだけのデータ処理に終らない、まさに"読み"の深さが違ってくる革新の機能です。

オペレーションに気をとられずに使いたい。だから書くように入力できるマウス。

話題の超小型入力装置マウスを採用しました。机の上で前後左右に動かすだけで、項目選択や、線・色・形の入力がごく簡単。キーボードにとらわれずに画面に集中できます。あなたの思考を助ける新時代のツールです。

どんどん発展するMZ-5500ソフト&ハード環境

〈新作オリジナルアプリケーション&システムソフトウェア〉■統合化ソフトTODAY…MZ-2Z014 標準価格 68,000円 ■日本語ワードプロセッサ(V1.1)……MZ-2Z025 標準価格 49,800円 ■ビジネス用BASIC-3…MZ-2Z017 標準価格 20,000円 ■日本語MS-DOS™*…MZ-2Z013

▶TODAYの使用にあたっては、オプションの漢字メモリが必要です(辞書ROMボードMZ-1R14の使用も可能)。必要ドライブ数は2基、必要メモリは256Kバイトです。また日本語ワードプロセッサの使用にあたってはオプションの漢字メモリMZ-1R10、辞書ROMボードMZ-1R14(JIS第二水準漢字ROM MZ-1R21の使用も可能)、および拡張ユニットMZ-1U05が必要です。また必要ドライブ数は2基、必要メモリは256Kバイトです。*MS-DOS™は米マイクロソフト社の登録商標です。

〈先進のオプション機器も続々登場〉■10.8MBハードディスク MZ-1F10近日発売■1MBミニフロッピーディスクドライブ MZ-1F13近日発売■80桁漢字プリンタ(24×24ドット) MZ-1P10近日発売■132桁漢字プリンタ(24×24ドット) MZ-1P11近日発売■辞書ROMボード(登録語数80,000語) MZ-1P14 標準価格 40,000円■JIS第二水準漢字ROM MZ-1R21近日発売■スタンダードフロッピーインターフェイス MZ-1E11 近日発売

16ビットパーソナルコンピュータ

MZ-5500シリーズ

MZ-5521	標準価格388,000円	●256KバイトRAM標準装備・ミニフロッピー2基内蔵
MZ-5511	標準価格288,000円	●128KバイトRAM標準装備・ミニフロッピー1基内蔵
MZ-5501	*標準価格218,000円	●128KバイトRAM標準装備

※MZ-5501は、従来のMZシリーズ用ミニフロッピーをお持ちの方のためのモデルです。
▲写真はMZ-5521。12型カラーディスプレイMZ-1D11、チルトスタンドMZ-1S08Gおよびマウス MZ-1X10はオプションです。
●画面は統合化ソフトTODAYのディスプレイ表示例でハメコミ合成です。

シャープ株式会社 本社 〒545 大阪市阿倍野区長池町22番22号☎(06)621-1221(大代表)
●お問い合わせ、資料請求は…シャープ㈱国内産機営業本部 〒162 東京都新宿区市谷八幡町8番地☎(03)260-1161(大代表)

資料請求券 MZ-5500 on I/MZ 5係

Oh!MZ ㈱日本ソフトバンク発行 Printed in Japan 定価480円 雑誌02179−5

昭和38年6月7日国鉄日本海縦貫線特別急行列車認可 昭和38年11月2日第三種郵便物認可